DOCUMENT RESUME

ED 297 765 IR 052 486

AUTHOR Dede, C. J.; And Others
TITLE Factors Shaping the Evolution of Electronic Documentation Systems. Research Activity No. IM.4.
INSTITUTION Research Inst. for Computer and Information Systems, Houston, TX.
SPONS AGENCY Lyndon B. Johnson Space Center, Houston, Tex.
PUB DATE Jan 88
NOTE 130p.
PUB TYPE Information Analyses (070) -- Reports - Research/Technical (143)

EDRS PRICE MF01/PC06 Plus Postage.
DESCRIPTORS Artificial Intelligence; Communications; *Computer Software; *Database Management Systems; Electronic Mail; Expert Systems; Heuristics; *Information Systems; Microcomputers; Programing Languages; *Space Sciences; *Systems Development; Technological Advancement; User Needs (Information)
IDENTIFIERS Cognitive Sciences; *Microelectronics; *National Aeronautics and Space Administration

ABSTRACT

The first of 10 sections in this report focuses on factors that will affect the evolution of Space Station Project (SSP) documentation systems. The goal of this project is to prepare the space station technical and managerial structure for likely changes in the creation, capture, transfer, and utilization of knowledge about the space station which might be useful to personnel involved in any aspect of SSP. Probable shifts in information system attributes are delineated, such as hardware capabilities, database techniques, knowledge representation formats, and user interface functionalities that are likely to occur over the total period of space station development. The next section describes the major types of SSP information stations that will be developed over the next several decades. The following six sections analyze leading edge developments external to the National Aeronautics and Space Administration (NASA) that are apt to influence the development of these information systems, with particular emphasis on their documentation capabilities. Two final sections summarize the implications for NASA of these external developments. The functional requirements for the Technical and Management Information System and the Software Support Environment are appended. (182 references) (CGD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED297765

FACTORS SHAPING THE EVOLUTION OF ELECTRONIC DOCUMENTATION SYSTEMS

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it

☐ Minor changes have been made to improve reproduction quality

- Points of view or opinions stated in this document do not necessarily represent official OERI position or policy

C. J. Dede
T. R. Sullivan
J. L. Scace

Research Activity No. IM.4

January, 1988

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Christopher J.Dede

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR05348

ricis
Research Institute for Computing and Information Systems
University of Houston - Clear Lake

T·E·C·H·N·I·C·A·L R·E·P·O·R·T

The RICIS Concept

The University of Houston-Clear Lake established the Research Institute for Computing and Information systems in 1986 to encourage NASA Johnson Space Center and local industry to actively support research in the computing and information sciences. As part of this endeavor, UH-Clear Lake proposed a partnership with JSC to jointly define and manage an integrated program of research in advanced data processing technology needed for JSC's main missions, including administrative, engineering and science responsibilities. JSC agreed and entered into a three-year cooperative agreement with UH-Clear Lake beginning in May, 1986, to jointly plan and execute such research through RICIS. Additionally, under Cooperative Agreement NCC 9-16, computing and educational facilities are shared by the two institutions to conduct the research.

The mission of RICIS is to conduct, coordinate and disseminate research on computing and information systems among researchers, sponsors and users from UH-Clear Lake, NASA/JSC, and other research organizations. Within UH-Clear Lake, the mission is being implemented through interdisciplinary involvement of faculty and students from each of the four schools: Business, Education, Human Sciences and Humanities, and Natural and Applied Sciences.

Other research organizations are involved via the "gateway" concept. UH-Clear Lake establishes relationships with other universities and research organizations, having common research interests, to provide additional sources of expertise to conduct needed research.

A major role of RICIS is to find the best match of sponsors, researchers and research objectives to advance knowledge in the computing and information sciences. Working jointly with NASA/JSC, RICIS advises on research needs, recommends principals for conducting the research, provides technical and administrative support to coordinate the research, and integrates technical results into the cooperative goals of UH-Clear Lake and NASA/JSC.

FACTORS SHAPING THE EVOLUTION
OF
ELECTRONIC DOCUMENTATION SYSTEMS

FACTORS SHAPING THE EVOLUTION OF ELECTRONIC DOCUMENTATION SYSTEMS

C.J. Dede
University of Houston--Clear Lake

T.R. Sullivan
TRS Research

J.R. Scace
University of Houston--Clear Lake

Preface

The Project Team: Dr. Christopher Dede (Principal Investigator) is a Professor at the Clear Lake campus of the University of Houston (UH-CL). His research specialities are technology forecasting/assessment, artificial intelligence, and strategic planning. Tim Sullivan and Jacque Scace (Research Associates) are Master's students in the Studies of the Future program at UH-CL. Tim did the primary research on KBMS, hardware, standards, and TMIS/SSE functionalities and wrote the initial drafts of those sections. Jacque was responsible for monitoring emerging research issues and for building the study's network of external resources.

Sponsorship: This project was conducted under the auspices of the Space Business Information Center at UH-CL. The Center's other work is directed to studying the needs for business information to aid in space commercialization. The support of Dr. Peter Bishop, Director of the Center, and his staff is gratefully acknowledged. In particular, Cissy Yoes (Graduate Research Assistant) was instrumental in the publication of this report.

The Research Institute for Computer and Information Systems (RICIS) at UH-CL provided the funding umbrella under which this research was coordinated. The mission of RICIS is to conduct, coordinate, and disseminate research on computing and information systems among researchers, sponsors, and users from UH-CL, NASA/JSC, the aerospace and computing industries, and other research organizations. Its Director, Dr. A. Glen Houston, can provide additional information about RICIS activities.

Funding: This research was supported by the National Aeronautics and Space Administration through Cooperative Agreement NCC 9-16 between the Lyndon B. Johnson Space Center and UH-Clear Lake.

Funding for the project was provided by the Spacecraft Software Division, Mission Support Directorate, Johnson Space Center. The guidance and vision of the NASA Monitor, Steve Gorman, has been vital to the success of this study.

The views and conclusions contained in this report are those of the authors and should not be interpreted as representative of the official policies, either express or implied, of NASA or the United States Government.

External Resources: Many of the researchers cited in this study were generous in providing reports and ideas to the project team. Of particular assistance through their sponsorship of site visits were:

Dr. Richard Burton, Xerox PARC
Dr. Bill Curtis, MCC
Michael Evans, Expertware
Dr. Peter Friedland, NASA/Ames
Dr. Linda Fox, Ford Aerospace & Communications
Dr. Paul Jensen, Lockheed Missles & Space
Dr. John Leggett, Texas A&M
Steve Lehto, Tektronix
Dr. Mark Lepper, Stanford
Dr. Matthew Lewis, CMU
Tim Merriam, WISE
Drs. Norm Meyrowitz and Nicole Yankelovitch, Brown University
Dr. Ray Nickerson, BB&N
Dr. Walt Scacchi, USC
Dr. Ben Shneiderman, University of Maryland
Dr. Peter Szolovits, MIT
Dr. Charles Thompson, Harvard

Executive Summary

This study focuses on factors which will affect the evolution of Space Station Project (SSP) documentation systems. The authors define "documentation" very broadly to encompass knowledge about the space station which might be useful to personnel involved in any aspect of SSP. This research delineates probable shifts in information system attributes such as hardware capabilities, data base techniques, knowledge representation formats, and user interface functionalities.

The goal of this project is to prepare the space station technical and managerial structure for likely changes in the creation, capture, transfer, and utilization of knowledge. By anticipating advances, the design of SSP information systems can be tailored to facilitate a progression of increasingly sophisticated strategies as the space station evolves.

A new paradigm for how to use information technologies is emerging. Originally, computers were seen as number crunching machines; with time, their data processing capabilities were recognized. Now, the strengths of integrated computer and telecommunications devices for all forms of individual and group symbolic manipulation are being explored.

The conceptualization of information technologies as symbolic manipulation devices is leading to research on how data and information can be converted to knowledge and wisdom. Past generations of management and technical information systems have used advances in hardware and software to increase the amount of data available, on the assumption that individual and institutional wisdom would thereby increase. In practice, however, high levels of data overwhelm people; they become unable to decide which information is important, to interconnect new information into existing knowledge, or to recognize overall patterns of meta-knowledge.

Future generations of advanced information systems will use increases in power to deliver environmentally meaningful, contextually targeted, interconnected data ("knowledge"). Leading edge research is focusing on how information systems can perform such a conversion of raw data. For example, the concept of a Knowledge Base Management System (KBMS) is emerging. Such a system would include traditional management functions for large shared databases (i.e. data models, query languages, semantic integrity maintenance, storage/search structures, update optimization, concurrency, security, error recovery, and distribution of data and processes). Added artificial intelligence features might encompass co-existing knowledge representation schemes; effective control structures for deductive, plausible, and inductive reasoning;

means for knowledge acquisition, refinement, and validation; explanation facilities; and dynamic human intervention.

Envisioning what a NASA Software Support Environment (SSE) workstation might resemble ten years from now is one method of summarizing the topics discussed in this report. The description below is an illustrative scenario of the generic components and capabilities of such a workstation.

> Sitting at a SSE access station in the year 1997, one is first struck by the most visible element of the workstation: the display device. The monitor is a twenty inch flat panel capable of presenting three-dimensional graphics, still and video images, and text in 4000 different colors. The screen resolution is close to four million pixels (2000 X 2000 lines of resolution); thus, it projects an image comparable to a high quality printed page. Telecommunications links are provided primarily by fiber optic networks, with satellite linkages a secondary system for long haul communication.
>
> The workstation offers easy connectivity with other computers, peripherals, and telecommunications devices, since its architecture is designed on the basis of the OSI and ISDN standards protocols and embedded microchips facilitate interconnectivity. This documentation station can also run programs designed for other computers via emulation.
>
> A dual-erasable optical disk system provides up to three gigabytes of secondary storage for mixed object knowledge bases. The system is equipped with a "megacard", which serves as both a means of security and a way to quickly upload or download information not captured on optical disk. The megacard is similar to a small plastic credit card and carries the user's security profile in a protected mode and up to two megabytes of erasable memory for other types of information defined by the user.
>
> The workstation has limited voice recognition (user specific, restricted vocabulary) and voice synthesis, coupled with a mouse (or other user-chosen input device) as major means of user communication with the device. The keyboard is used for more complex entry requirements. The User Interface Management System (UIMS) is designed to offer intelligent assistance (via embedded coaches and tutors) to novices, but also incorporates powerful command sets for experienced users.
>
> Interface functionalities available include direct manipulation, mimetic environments, limited "artificial realities," advanced input devices (such as the DataGlove), and "microworlds." User productivity skills such as "linearizing" (performing several tasks while switching among them), task mapping, and "metacognition" ('thinking about thinking' to see patterns of suboptimal performance) are empowered via "cognitive audit trails," "consciousness sensors," and intelligent, semi-autonomous agents. Through the workstation's enabling computer-supported cooperative work, the user's extrinsic motivation is enhanced via cooperation, competition, and recognition. Intrinsic motivation is built via creating an task environment which maximizes challenge, fantasy, curiosity, and control.

The hardware architecture is driven by a distributed parallel processor with 25 megabytes of local random access memory rated at 20 MIPS. The parallel processor contains specialized embedded chips for image/graphic processing and voice recognition and can function as a linked distributed node on the network to which it is connected. Thus, it can provide expanded computing power for large scale demands within its local area network. This amount of power is capable of 100 LIPS (Logical Inferences Per Second) per MIP (Million Instructions Per Second) over an object knowledge base of 10^{10} objects, at a clock speed of 80 megahertz.

Such a system allows the user to interact with large knowledge bases, which contain the equivalent of universally quantifiable statements (e.g. "people of middle age are careful") rather than simply pieces of data, such as "Mr. Lee is 43 years of age." A Knowledge Base Management System (KBMS) handles deductive reasoning/search, inductive reasoning, explanation, knowledge refinement and validation, and automatic classification of knowledge in a manner transparent to the user.

The workstation supports alternative knowledge representation formats, such as "hypermedia." This use of associational networks allows the implementation of sophisticated systems for conceptual exploration, retrieval, training, retention, group collaboration, customization, and revision. Navigational aids help to minimize problems of disorientation and cognitive overhead, while difficulties with combinatorial explosion and collective communications are lessened through the use of instructional design strategies tailored to intelligent tutoring systems.

The workstation is connected as a node in the SSE network and has both telephonic and electronic mail links to all NASA information systems. Attached is a "black box" which allows the addition of specialized manipulation devices. Connecting an A/V camera enables the workstation to become a fully operational teleconferencing station. Real time digitization and storage of video data up to thirty frames per second is possible.

By windowing the screen, the user can participate concurrently in a teleconference while manipulating information with other participants in a cooperative work environment. Collaborative design (surfacing collective assumptions, resolving conflicts, developing shared models of a task), shared problem solving (text sharing, project management, collaborative authoring), and group decision support are enabled. WYSIWIS (What You See Is What I See) provides a common visual representation which can be manipulated by all participants.

Topics recommended for further exploratory research projects include:

- a user-based, detailed description of a future SSE workstation
- a compilation of NASA's internal SSE research on knowledge bases
- constructing a prototype hypertext SSE documentation system
- building a simulated SSE interface with advanced capabilities
- testing knowledge transfer techniques' effectiveness for training
- developing collaborative design capabilities for the SSE
- synthesizing research on the management of knowledge systems
- synthesizing research on software engineering

Table of Contents

Preface i

Executive Summary iii

Introduction 1
- Scope of this Research 1
- Rationale for this Study 2
- Strategies for Resolving these Problems 3
- An Image of the Future 6
- Summary 8

An Overview of Space Station Information Systems 9
- The Role of Documentation Systems in SSP 10

Progressing from DBMS to KBMS 12
- Advantages of KBMS 12
- Emerging Themes and Evolution 14
- Development Issues and Illustrative Projects 17
- Summary 19

Alternative Knowledge Representation Approaches 21
- The Concept of Hypermedia 21
- Current Systems 25
- Operational Challenges 32
- Evaluating the Usefulness of Hypermedia 34
- Other Potential Representation Formats 35
- Summary 36

Advanced User Interface Capabilities 38
- Goals of User Interface Design 38
- Illustrative Promising Directions 39
- Summary 47

Computer-Supported Cooperative Work 48
- Collaborative Design Technology 48
- Organizational Interfaces 50
- Summary 52

The Evolution of Information System Hardware 54
- Problems of Rapid Hardware Advances 54
- Managing Hardware Evolution 56
- Strategic Planning for Hardware Developments 57
- Advances in Computer Architectures 58
- Advances in Secondary Mass Storage Devices 61
- Advances in Telecommunications Technology 64
- Hardware Integration and Long Range Evolution 69
- Timetable 71
- Summary 73

Standardization, Compatibility, and Connectivity 75
Standards Organizations 76
Telecommunications Standards 78
Computer and Information Processing Standards 79
Summary 80

Organizational Impacts of Information Intensive Environments 82
Consequences of Implementing Advanced Documentation Systems 83
Heuristics for Improving Implementation 86
Summary 88

High Leverage Topics for Further Exploration 89
Illustrative Research Projects 90
Conclusion 94

References 96

Appendix

Functional Requirements for TMIS and SSE 110

Introduction

The National Aeronautics and Space Administration (NASA) is now devoting substantial resources to the development of a space station. To accomplish this task, managers at every level of the Space Station Project (SSP) must coordinate the work of their teams with many related activities which are geographically and departmentally dispersed. Such a large scale development structure necessitates the creation of multiple sophisticated technical and managerial information systems. Since historically NASA's decentralized information resources have not been well integrated, a strategic plan for Automated Information Management (AIM) has been created to guide the evolution of all information systems, including SSP-related activities.

Four SSP information systems for which this study will be useful are the Technical and Management Information System (TMIS), the Software Support Environment (SSE), the Program Support Communication Network (PSCN), and the Space Station Information System (SSIS). TMIS will be an integrated system which provides automated information and communication resources to technical and managerial personnel at all levels of NASA and its customers, contractors, and international partners. The SSE will be an independently developed system which supplies the software engineering tools and techniques for life cycle development, integration, and management of space station operational software. After the initial phases of TMIS and SSE are completed, SSIS (the information and communications system for space station operations) will be developed. The PSCN will provide the data, voice, and video communications devices which interconnect TMIS, SSE, and SSIS sites.

Scope of this Research

This study focuses on the factors which will affect the evolution of SSP documentation systems, using the SSE as a paradigmatic example. The authors define "documentation" very broadly to encompass knowledge about the space station which might be useful to personnel involved in any aspect of SSP. The primary objective of this research is to delineate, over the total period of space station development, probable shifts in information system attributes such as hardware capabilities, data base techniques, knowledge representation formats, and user interface functionalities.

The goal of this project is to prepare the space station technical and managerial structure for likely changes for the creation, capture, transfer, and utilization of knowledge. Some of these shifts will be opportunity-driven (e.g. more powerful hardware, better software engineering techniques); other changes will be forced because of problems with the scale, uncertainty, and complexity of the station's development. By anticipating advances, the design of SSP information systems can be tailored to facilitate an orderly progression of increasingly sophisticated strategies as the space station evolves.

Project outcomes include 1) a synthesis of advanced research on documentation, 2) a network of external resources doing leading edge work, and 3) a menu of high leverage topics which merit further exploration. To limit the scope of this study, its focus is targeted to concepts that are practical within a mid-range time frame, specific to station needs, and not currently major components of SSP information system development. This final report contains the research synthesis and a menu of potential funding activities. The briefings delivered by the authors will aid in linking the external resources network to relevant personnel involved in space station development.

This study is being conducted at a time when the attributes, architectures, and roles of SSP information systems are still in flux. Some details in the functional requirements for these systems, while accurate when written, may therefore eventually alter. The overall conclusions of this report should still hold, however, as the major decisions about SSP configurations have been made.

Rationale for this Study

Imagine the space station as a human being. The physical structure of the station and its machines are the bones and muscles of its body; software is its nervous system; human users, its brain. Documentation provides the connection between the station's brain, body, and nervous system; if this link falters, paralysis will rapidly occur.

Forces certain to affect SSP documentation systems include:

- the rapid evolution of technical workstations (i.e. new capabilities, lower prices, standardization, issues of connectivity and compatibility with older equipment)
- new attributes for conventional large data base management systems (e.g. templates, alternative options for mass digital storage, integration of multi-media information formats)
- a more important role for documentation as SSP's central "organizational memory," since the life cycle of the space station encompasses several generations of engineering and managerial personnel
- a gradual shift in the user base for SSP documentation systems from R&D to operational applications

Other possible developments during the next decade are 1) the widespread adoption of "hypermedia" (nonlinear storage of multi-media information) as an alternative knowledge representation system and 2) the emergence of advanced user interface capabilities (such as sophisticated environments for computer-supported cooperative work, "artificial realities" to facilitate design explorations, and intelligent tutoring systems).

All these innovations will drive a shift to an information-abundant environment which has implications for SSP's implementation strategies, management policies, personnel training, and organizational culture. Problems of knowledge creation, capture, transfer, and utilization which currently impede the creation of large scale technological systems such as the space station include:

1) An overwhelming amount of interconnected information is vital to task performance.
2) Increasing the data personnel must absorb can overwhelm their ability to synthesize knowledge from this information.
3) Difficulties in software production, integration, maintenance, and evolution are a key bottleneck.
4) Implementing new approaches in an established design structure is very challenging.

Conventional documentation strategies have failed to resolve these problems.

Moreover, the space station project has unique characteristics which make these issues especially acute. These include attributes of *scale* (thousands of personnel, distant geographic links, terabytes of information flow); *complexity* (integration of multi-media information formats, networking among diverse kinds of workstations, different types of users, closely coupled design decisions, the necessity of continuous operation); and *uncertainty* (extended time horizons, rapid evolution of technological base, "proof of concept" tasks, an unstable political and economic context), as well as the fact that the station is a "life-critical" system. As the most complex technological artifact ever created, the station offers a very challenging testbed for documentation approaches.

Strategies for Resolving these Problems

This study is based on the assumption that the trends and issues above require more functionality from documentation than passive storage of data. A computer's internal random access memory (RAM) can be augmented by disk-based virtual memory, greatly extending its capabilities. Similarly, personnel involved in space station development need a technology-based "knowledge structure" which unobtrusively amplifies human cognition and memory.

A new paradigm for how to use information technologies is emerging. Originally, computers were seen as number crunching machines; with time, their data processing capabilities were recognized. Now, the strengths of integrated computer and telecommunications devices for all forms of individual and group symbolic manipulation are being explored.

The conceptualization of information technologies as symbolic manipulation devices is leading to research on how data and information can be converted to knowledge and wisdom. Common usage of these terms conveys a sense of increasing complexity and utility, but a rigorous delineation of their subtle differences is much more difficult. As a simplified category system for the purposes of this study, "data" will be defined as input gathered through the senses; and "information" as integrated data which denotes a significant change in the environment. (Anthropologist Gregory Bateson [1975] defined "information" as "any difference that makes a difference.")

Information is converted to "knowledge" by interconnecting it with known concepts and skills as part of achieving a goal. {Note that knowledge has an attribute of purpose, which implies the existence of an intelligent agent (human or computational) in transforming information into knowledge.} "Wisdom" is knowledge about knowledge ("meta-knowledge"). To illustrate, for a software programmer, the existence of a new primitive procedure in the programming language would be data, an understanding of what new functions it adds to the language would be information, comprehension of how to use the primitive in coding would be knowledge, and mastery of when to use the primitive and of its effects on overall programming style would be wisdom.

Past generations of management and technical information systems have used advances in hardware and software to increase the amount of data available, on the assumption that individual and institutional wisdom would thereby increase. In practice, however, high levels of data overwhelm people; they become unable to decide which information is important, to interconnect new information into existing knowledge, or to recognize overall patterns of meta-knowledge. Future generations of advanced information systems will use increases in power to deliver environmentally meaningful, contextually targeted, interconnected data ("knowledge"); and leading edge research is focusing on how information systems can perform such a conversion of raw data.

For example, the concept of a Knowledge Base Management System (KBMS) is emerging. Such a system would include traditional management functions for large shared databases (i.e. data models, query languages, semantic integrity maintenance, storage/search structures, update optimization, concurrency, security, error recovery, and distribution of data and processes). Added artificial intelligence features might encompass co-existing knowledge representation schemes; effective control structures for deductive, plausible, and inductive reasoning; means for knowledge acquisition, refinement, and validation; explanation facilities; and dynamic human intervention [Brodie & Mylopoulos, 1986].

As an illustration, emerging software lifecycle strategies which are being explored as possible new approaches to knowledge creation, capture, transfer, and utilization include:

- documentation integrated into the software production process as a means of reducing life cycle maintenance costs
- object-oriented documentation systems to maximize software reusability
- hypermedia with embedded inference engines for propagating changes through the code structure in order to reinforce software engineering strategies
- documentation which captures the rationale and mental models underlying design decisions to improve the overall development process
- intelligent, active documentation systems as a method of reducing the negative aspects of interleaving task performance with storing information about the task
- documentation systems which include instructional features to enhance the training of new personnel
- documentation development mechanisms structured to provide a "cognitive audit trail" for management

Prototype systems which exemplify these concepts are discussed in the research synthesis sections later in this report.

However, a full review of recent software engineering advances is beyond the scope of this study. Another document comparable in size would be required to discuss the numerous promising research themes emerging. Resources for SSE developers and others interested in this topic include:

- The entity-relationship model research directed by Charles McKay at the Software Engineering Research Center, University of Houston--Clear Lake [McKay, 1987]
- The paradigm for life cycle software activities created by Michael Evans [Evans & Marciniak, 1987]
- The projects of the Software Productivity Consortium [Yudkin & Nidiffer, 1987]
- Work on WISDM at the Western Institute of Systems Engineering
- Artificial intelligence advances in automatic programming, theorem proving, program construction via transformations, specification techniques, intelligent assistants, and knowledge representation [Rich & Waters, 1986]
- David Parnas' reflections on the limitations of software design and engineering [Parnas, 1985; Parnas & Clements, 1986]
- Work toward project master data bases for software engineering environments [Penedo & Stuckle, 1985]

- Emerging developments in knowledge abstraction [Abbott, 1987]
- Lockheed's work on its proprietary software engineering system, PLEXUS, under the leadership of Dr. Paul Jensen
- Progress toward automating the software development cycle [Frenkel, 1985]

Shaw [1985] presents a good overview of the next generation of challenges software engineering efforts must address.

An Image of the Future

Envisioning what a SSE workstation might resemble ten years from now is one method of previewing the topics discussed in the sections following. The description below is an illustrative scenario of the generic components and capabilities of such a workstation. (The task of presenting specific software engineering tools and functionalities of an SSE workstation ten years from now would require a separate, extended study, as discussed in the **High Leverage Topics for Further Exploration** section.)

> Sitting at a SSE access station in the year 1997, one is first struck by the most visible element of the workstation: the display device. The monitor is a twenty inch flat panel capable of presenting three-dimensional graphics, still and video images, and text in 4000 different colors. The screen resolution is close to four million pixels (2000 X 2000 lines of resolution); thus, it projects an image comparable to a high quality printed page. Telecommunications links are provided primarily by fiber optic networks, with satellite linkages a secondary system for long haul communication.
>
> The workstation offers easy connectivity with other computers, peripherals, and telecommunications devices, since its architecture is designed on the basis of the OSI and ISDN standards protocols and embedded microchips facilitate interconnectivity. This documentation station can also run programs designed for other computers via emulation.
>
> A dual-erasable optical disk system provides up to three gigabytes of secondary storage for mixed object knowledge bases. The system is equipped with a "megacard", which serves as both a means of security and a way to quickly upload or download information not captured on optical disk. The megacard is similar to a small plastic credit card and carries the user's security profile in a protected mode and up to two megabytes of erasable memory for other types of information defined by the user.
>
> The workstation has limited voice recognition (user specific, restricted vocabulary) and voice synthesis, coupled with a mouse (or other user-chosen input device) as major means of user communication with the device. The keyboard is used for more complex entry requirements. The User Interface Management System (UIMS) is designed to offer intelligent assistance (via embedded coaches and tutors) to novices, but also incorporates powerful command sets for experienced users.

Interface functionalities available include direct manipulation, mimetic environments, limited "artificial realities," advanced input devices (such as the DataGlove), and "microworlds." User productivity skills such as "linearizing" (performing several tasks while switching among them), task mapping, and "metacognition" ('thinking about thinking' to see patterns of suboptimal performance) are empowered via "cognitive audit trails," "consciousness sensors," and intelligent, semi-autonomous agents. Through the workstation's enabling computer-supported cooperative work, the user's extrinsic motivation is enhanced via cooperation, competition, and recognition. Intrinsic motivation is built via creating an task environment which maximizes challenge, fantasy, curiosity, and control.

The hardware architecture is driven by a distributed parallel processor with 25 megabytes of local random access memory rated at 20 MIPS. The parallel processor contains specialized embedded chips for image/graphic processing and voice recognition and can function as a linked distributed node on the network to which it is connected. Thus, it can provide expanded computing power for large scale demands within its local area network. This amount of power is capable of 100 LIPS (Logical Inferences Per Second) per MIP (Million Instructions Per Second) over an object knowledge base of 10^{10} objects, at a clock speed of 80 megahertz.

Such a system allows the user to interact with large knowledge bases, which contain the equivalent of universally quantifiable statements (e.g. "people of middle age are careful") rather than simply pieces of data, such as "Mr. Lee is 43 years of age." A Knowledge Base Management System (KBMS) handles deductive reasoning/search, inductive reasoning, explanation, knowledge refinement and validation, and automatic classification of knowledge in a manner transparent to the user.

The workstation supports alternative knowledge representation formats, such as "hypermedia." This use of associational networks allows the implementation of sophisticated systems for conceptual exploration, retrieval, training, retention, group collaboration, customization, and revision. Navigational aids help to minimize problems of disorientation and cognitive overhead, while difficulties with combinatorial explosion and collective communications are lessened through the use of instructional design strategies tailored to intelligent tutoring systems.

The workstation is connected as a node in the SSE network and has both telephonic and electronic mail links to all NASA information systems. Attached is a "black box" which allows the addition of specialized manipulation devices. Connecting an A/V camera enables the workstation to become a fully operational teleconferencing station. Real time digitization and storage of video data up to thirty frames per second is possible.

By windowing the screen, the user can participate concurrently in a teleconference while manipulating information with other participants in a cooperative work environment. Collaborative design (surfacing collective assumptions, resolving conflicts, developing shared models of a task), shared problem solving (text sharing, project management, collaborative authoring), and group decision support are enabled. WYSIWIS (What You

See Is What I See) provides a common visual representation which can be manipulated by all participants.

Such an environment would have far more powerful capabilities for knowledge creation, capture, transfer, and utilization than equivalent workstations today. A timetable for when many of these functionalities may develop is given at the end of the section on **The Evolution of Information System Hardware.**

Summary

Initiatives in artificial intelligence and cognitive science--coupled with advances in user interface functionalities, the power of technical workstations, and the synthesis of computers and telecommunications--are creating new conceptions of knowledge creation, capture, transfer, and utilization. No matter how sophisticated in current terms the space station's initial documentation strategy may be, its approach will be hopelessly obsolete decades later when the station is fully operational. Designing the early stages of SSP information systems to have the flexibility to incorporate new approaches as they mature is a vital priority.

While the report discusses implications for "electronic documentation," the research in this study can be viewed from different perspectives depending on the orientation of the reader. For example, those familiar with configuration management and control issues will see many applications to that topic. The creation, capture, transfer, and utilization of knowledge is a generic theme which underlies many types of SSP activities; readers are encouraged to draw implications based on how they use information in task performance.

The remainder of this report is grouped into several parts. First, the next section (**An Overview of Space Station Information Systems**) describes the major types of SSP information systems which will be developed over the next several decades. Then, the six following sections (**Progressing from DBMS to KBMS; Alternative Knowledge Representation Approaches; Advanced User Interface Capabilities; Computer-Supported Cooperative Work; The Evolution of Information System Hardware; Standards, Compatibility, and Connectivity**) analyze leading edge developments external to NASA likely to influence the development of these information systems, particularly focusing on their documentation capabilities. Two sections (**Organizational Impacts of Information Intensive Environments, High Leverage Topics for Further Exploration**) summarize the implications for NASA of these external developments. Finally, **References** and an Appendix (**Functional Requirements for TMIS and SSE**) conclude the report.

An Overview of Space Station Information Systems

This section of the study briefly describes the major types of space station information systems which will be developed over the next several decades. In order to understand the role of these systems in overall NASA strategy, the evolution of NASA information system technology from a historical perspective is reviewed below.

The National Aeronautical and Space Administration is composed of ten major entities having varying responsibilities and located in different regions of the United States. There are three types of NASA entities: field centers (Marshall, Johnson, Goddard, and Kennedy), research centers (Ames, Langely, and Lewis) and laboratories (the Jet Propulsion Lab and the National Space Technology Lab). These entities are managed through a decentralized organizational structure (NASA HQ), which is located in Washington D.C.

Historically, the centers and labs have developed their information resources based upon the needs of each institution and limited to that individual site. These information systems were composed of manual and automated systems which were characteristically not integrated--due both to the unavailability of the necessary technologies and the lack of an overarching strategic incentive. In particular, integration has been unavailable both intra-center and inter-center for information systems designed to support technical and management processes for research and development.

Efforts are underway to develop strategies which better utilize NASA information resources. These strategies are embodied in the "Automated Information Management" (AIM) plan [NASA, 1986g]. This plan calls for the development of information system technology which strengthens and complements a decentralized management approach; improves information utility and delivery; provides an architecture which is flexible, user oriented and growth oriented; and eliminates duplication and wasteful use of resources.

The Space Station Project (SSP) promises to be one of NASA's most ambitious undertakings. Encompassing a budget originally estimated at $8 billion and which currently ranges from $25 billion to $30 billion, SSP will require the combined efforts of 4500 independent contractor organizations and international partners such as Canada, Japan, and the European Space Agency.

SSP is composed of numerous subsystems. The engineering paradigm of reductionism has been employed to design the Space Station and all its components. This paradigm, simply stated, breaks the overall structure into many small subsystems and parcels them out to various participating organizations. As they are completed, these subsystems will be reconfigured back into the total architecture. Major challenges of this approach include ensuring that the various components are designed to fit together and accomplishing this integration. Sophisticated information systems are vital to meeting these challenges.

This study examines the future of knowledge creation, capture, transfer, and utilization for SSP; as paradigmatic examples, the documentation issues affecting two particular information systems are traced. The Technical Information Management System (TMIS) and the Software Support Environment (SSE) are critical information systems needed to support development of SSP; details on the required functionalities for these systems are given in the **Appendix**. Documentation issues have been selected because this type of knowledge serves as the primary communications medium linking specialists working on different aspects of the development, use and maintenance of the space station system [Garg, Jazzar, & Scacchi; 1986a].

The Role of Documentation Systems in SSP

Historically, two characteristics of very large scale engineering projects such as SSP are 1) documentation has the role of providing vital links among all aspects of research, development and operations; and 2) this role is rarely accomplished well. One objective of this study is the identification of likely SSP documentation challenges. Problems which occur in the documentation of complex, large scale research and development projects which extend over long periods of time stem from four types of issues:

- Extended temporal duration
- Scale: information volume, access, distribution, updating
- Information complexity
- Cooperative working environments involving many individuals with a wide range of documentation skills, and needs

Typically, documentation problems are characterized by incompleteness, inconsistency, limited contextual usage (different individuals have varying documentation needs for the same system), limited traceability which restricts tracking common development treads through life cycle stages (e.g. a change in a line of software code could affect hundreds or thousands of pages of supporting documentation), and inadequate management of multi-media data formats and document revisions [Garg, Jazzar, & Scacchi, 1986b].

In order to understand the roles of TMIS and SSE and the various information systems with which these interface, brief descriptions of these component systems follow. (A functional capability overview on TMIS and SSE is provided in greater detail later in this section.) TMIS is an integrated system of technical and management processes, automated data processing hardware and software, communications, procedures, and people supporting the design, development and management of the Space Station

Program [Harlan, 1986]. The overarching AIM and SSP strategies require that TMIS interface effectively with other support systems in both SSP and NASA Institutional. For example, three other key SSP systems to which TMIS will interface are the Space Station Information System (SSIS), the Software Support Environment (SSE) and the Program Support Communication Network (PSCN).

SSIS is the set of hardware, software, and interconnections required for the development, simulation, training, operation, and maintenance of the SSP elements, associated institutional resources, and international and customer sectors [NASA, 1986f]. The functions supported by the SSIS include prelaunch checkout; mission management; scheduling; control; support of onboard and ground operational software development; and the acquisition, transmission, recording, processing, accounting, storage, and distribution of data produced by the SSP space and ground operational elements.

The SSIS is intended to serve the users of the SSP, including Space Station onboard crew, ground support operators, researchers, and commercial customers. Selected TMIS capabilities (tools) such as electronic mail, configuration management and document generation, storage, and retrieval will be utilized by SSIS. SSIS phasing is expected to follow implementation of TMIS.

The Program Support Communication Network (PSCN) [NASA, 1986e] is the communications hardware, software, and transmission medium that will be used to interconnect, for the transfer of data, voice, and video, NASA-wide facilities including HQ, all centers and labs, other field installations, and contractors [NASA, 1986b]. Although PSCN is part of a separately managed contractual effort, TMIS will make maximum use of PSCN capabilities. Thus, PSCN will serve as a major portion of the TMIS communications component.

The Software Support Environment (SSE) [NASA, 1986b] is the collection of software, procedures, standards, hardware specifications, documentation, policies, and training materials which, when implemented, will provide the environment used for the life cycle management of operational SSP software. The SSE is intended to be the mechanism which supplies the software tools and environment to support station software development. SSE is expected to interface with TMIS and the PSCN closely; its development and implementation are concurrent with TMIS.

The sections of this study which follow describe emerging approaches to managing knowledge creation, capture, transfer, and utilization in projects of this scale, duration and complexity. These incipient strategies fall short of providing an immediate, robust, comprehensive solution for TMIS and the SSE. Rather, they are potential elements of an overall future solution of which this study provides a partial assessment.

PROGRESSING FROM DBMS TO KBMS

Several emerging themes in the development of Knowledge Based Management Systems (KBMS) and their integration with Data Base Management Systems (DBMS) are discussed below. The central focus of this section is to summarize recent research on comprehensively defining a bounded aspect of reality (a knowledge domain) and linking it to machine-based reasoning capabilities. The section following, **Alternative Knowledge Representation Approaches**, builds on this background in its synthesis of research on new representational architectures for knowledge.

Implementing a knowledge base requires 1) a complete description of the knowledge domain and the objects within it, and 2) the ability of the KBMS to make inferences about the relationships of the domain objects. For instance, capturing the descriptive characteristics of "ownership," "causality," "dimension," and "time" in a conventional DBMS is not possible. KBMS technology would provide the functionality to reason about these important types of attributes within the context of a data base.

DBMS technology does a good job of managing "data" (an isomorphic representation of domain objects as described by alphabetical or numeric character strings in a two dimensional matrix of data fields represented by rows and columns). Additional relationships can then be defined which link matrices together, creating logical relationships. Within current DBMS technology, however, the overarching entity described by the various descriptive data elements is not managed as a coherent object with certain characteristics ("knowledge"), but rather each piece of data is treated individually.

To illustrate, envision a conventional DBMS which documents a parts list of the Space Station Life Support System. Such a system would include data element descriptions (such as part number, name, weight, cost, manufacturer, and a textual functional description)--but these chunks of information have no inherent meaning to the DBMS. All the intelligence needed for problem solving must reside in the user, who needs to be very sophisticated in querying this system in order to extract information about the relationships among these descriptors.

Advantages of KBMS

In contrast, imagine the following scenario of working with an hypothetical KBMS containing the same data. Suppose that you are a maintenance engineer at a NASA facility. The Space Station, now operational, is experiencing problems with the Life Support System, which was designed and built three years before you joined NASA. Thus, you must rely heavily upon documentation to solve this problem.

Attempting to understand what has happened, you collect a set of pathological symptoms exhibited by the Life Support System. You then gain access to an electronic documentation knowledge base and describe in restricted natural language the problem's symptoms and your tentative diagnostic hypotheses. The documentation system considers your query, delineates the hardware and software entities in the Life Support System which may be involved, and identifies four potential culprits for further investigation: two physical components and two software modules.

You request a computer-generated graphical representation of "Part A" and a three-dimensional mapping of how this part fits with others attached to it (producing this requires the KBMS to use executable dimensional semantics and constructive solid geometry). Subsequent requests you make of the knowledge system include an interpretation of what "Part A" does to "Part B" when the Life Support System is operating, the means by which the Life Support System software directs this interaction, and the sequence of manipulating these parts on assembly; these require qualitative causal reasoning about temporal events. The capabilities of capturing and processing such pieces of knowledge about a domain are the foundation of KBMS function.

A knowledge based approach is important to SSP documentation because such systems potentially offer a means for resolving three major issues:

- The complexity and volume of knowledge generated by SSP
- The changing role of documentation as the project progresses
- The loss of institutional memory over time

The complexities of developing the Space Station will require information technologies which are beyond the functionalities and integration capabilities of current DBMS technology. For example, one critical aspect of documentation is the need to capture, in a digital format, a comprehensive description of the knowledge created during the SSP development process. Such knowledge is composed of many elements; these lie on a continuum in which more detailed descriptions of reality come at the cost of increasing complexity. KBMS technology offers approaches for capturing this complexity which go beyond what is possible by storing "data."

EVOLVING DOCUMENTATION CHARACTERISTICS

Present Work	Flight Verification	Space Station Operation
IOC'	IOC-5	Evolutionary Phase

<===>

Authors: more............................less......................................
# of Users: more..less...................
Type of Users: Design engineers..................................Flight Specialists.......
Purpose: Knowledge capture/research............................Maintenance.....
Setting: Development....................................Operation.................
Process: Design.................Troubleshoot.....................Train..............
Volume: low.............................high........................medium........

As indicated, the role of documentation will change as SSP moves through its lifecycle. During this evolution, factors such as the number of authors and users, the setting, and the major purposes of NASA documentation systems will alter. Embedded KBMS technology could be of value, as one of its strengths is the ability to create contextual frames for using data based upon immediate needs.

Another virtue of a KBMS approach is its capability to capture human expertise. Institutional memory loss is becoming recognized as a major cost of large scale engineering efforts which occur over extended time frames. This is a crucial concern, as the underpinning of a project such as SSE is the collective knowledge of the individuals working on the project. Personnel turnover, inevitable in a long range endeavor, erodes this knowledge and can degrade task performance by forcing the same problem to be solved again and again.

Helping with such collective amnesia is an important role of documentation, but requires capturing a comprehensive representation of task performance in the development and maintenance of complex SSP software and systems. KBMS has the potential to capture, retain, manipulate, and disseminate this expertise, providing such benefits as building a trans-generational project memory; reducing the costs of development, maintenance, and training; and minimizing the negative effects of personnel changes over time.

Emerging Themes and Evolution

KBMS as a technology is diverse in terms of its definitions and architectures. As an illustration, alternative terminology to "KBMS" includes "expert data base systems," "intelligent data base systems," and "knowledge based systems." Four types of development paths from DBMS to KBMS are [Kerschberg, 1986]:

- an expert system loosely coupled with a data base system
- a data base management system enhanced with reasoning capabilities to perform knowledge-directed problem solving
- a logic programming system, or a knowledge representation system, enhanced with data base access and manipulation primitives
- an intelligent natural language interface to a data base system

These emerging configurations for integrating DBMS technology with knowledge representation are likely to be a sequence of evolutionary steps. For example, systems consisting of loosely coupled expert systems plus DBMS technology will probably appear sooner than systems which utilize tightly coupled logic programming plus DBMS technology. In the former case, the architecture involves linking discreet heterogeneous capabilities and information formats together under a homogeneous umbrella (the expert system). Such an approach requires a conventional data base to supply alphanumeric data to an inference engine coupled with rules or frames. Information formats such as images and graphics would be contained in their own unique file system, and specialized processors linked to the DBMS would handle all manipulations.

In the case of the logic programming system, however, complex information structures must be stored as well as records. The system has to blend deduction and procedural constructs into information management and to amalgamate intensional (schema) information with extensional (instance) information [Parker et al, 1986]. Such a system poses considerably more difficult technical problems and can be expected to develop more slowly.

Historical Evolution

DBMS technology has its early roots in problems which stemmed from the 1960's methodology of developing data processing software within an organization. (This was also known as "programming in the small.") Typically, an institution was confronted with problems "A,B,C" and data (highly structured information which could be contained and manipulated in a fixed field type architecture) associated with each individual problem. The conventional method of handling this situation was to develop unique, automated programs "1,2,3" for solving these specific problems.

PROBLEM "A" + DATA + PROGRAM 1 = OUTPUT 1
PROBLEM "B" + DATA + PROGRAM 2 = OUTPUT 2
PROBLEM "C" + DATA + PROGRAM 3 = OUTPUT 3 ....

Over time, the result was multiple programs having no design consistency, containing redundant and often inconsistent data, and generating mutually incompatible output. No easy way could be found to integrate these various programs, both because they had been developed in a vacuum with little thought given to overlap and because they were built using inconsistent methods and tools.

The introduction of DBMS technology in the 1970's improved this situation. DBMSs provided a common set of data processing tools and architectures to build data applications; a query language which made information retrieval easier and far more flexible; and the capability to capture data only once, but then to use it in a variety of programs. The results were data integrity and consistency between differing programs, easier accessibility, and a faster development lifecycle for applications.

However, DBMS technology did not significantly affect the problem of capturing different, more abstract information formats; these systems were still relegated to managing highly structured "data." Modeling real world phenomena requires much more than columns and rows of alphanumeric charter strings. The late 1970's to mid-1980's saw the evolution of office automation and the concept of managing "information" rather than just "data". Much of this shift was driven by the development of word processors, since these systems began capturing the next genre of information, text, into a magnetic medium. {Text differs from data in that its representational format is sentences and paragraphs contained in a document (as opposed to the columns and rows in a matrix).}

The current age of information processing and office automation is oriented to innovations such as desktop publishing (which merges text with graphics), facsimile systems for transmitted documents as an image, and full text retrieval (indexing complete documents on a word-by-word basis, allowing for complex queries of words and phases with Boolean and proximity logic). However, as currently designed, all these systems require unique file structures.

Most organizations have implemented a variety of devices and applications to resolve problems which deal with the management of text; as a result, the "data processing" problems of the early 1970's have reappeared in a slightly different form. The dissemination of multiple applications programs which employ different text management solutions has resulted in little integration both between the various text management systems and with the DBMS applications institutions already in place.

The mid-1980's saw the emergence of text management strategies designed to address this lack of integration between the various text systems and DBMS applications. For example, Ford Aerospace and Communications, under the leadership of Dr. Linda Fox, has developed an impressive system for integrating text and graphics files produced with multiple applications on devices from different vendors. System vendors have introduced strategic products such as IBM's DISOSS™, a software package aimed at integrating numerous non-compatible IBM hardware/software products so that a document created on any device under the DISOSS umbrella will be accessible and revisable on any other device.

Another such strategy is mainframe-based, text and data base management systems, such as Data Retrieval Corporation's TextDBMS™. This is a set of mainframe-based software components constructed around a textual data base architecture. The TextDBMS system provides full text search and retrieval, text processing, application development tools, linkage with existing DBMS technology to provide integrated management of "text" and "data," publishing components as an output, and a PC/mainframe link with automated conversion aids (which allow PC documents to be transferred in a standard revisable document format).

These types of systems, which are still evolving, provide benefits similar to DBMS technology. They offer commonality and integration for the office automation environment, allowing "information" to be managed under a set of common automation tools and facilitating linkage with existing DBMS systems. A likely next step will be linking the management of images and graphics as integrated objects in the data base. Lesk [1984] provides a good overview of ongoing research in this area.

Now, a similar evolution is beginning in the development of expert systems and knowledge based management systems. While a DBMS represents and manages facts, a KBMS describes and operates on classes of objects [Brachman & Levesque, 1986]. For example, Knowledge Bases contain the equivalent of universally quantifiable statements (e.g. "People of middle age are careful"), while data bases have the equivalent of atomic assertions (e.g., "Mr. Lee's age is 43 years"). The concept of combining DBMS functional strengths (such as data modelling, query/browsing, concurrency, error recovery, security, definition, maintenance, and search and update optimization) with more advanced capabilities (knowledge representation schemes, deductive reasoning/search, inductive reasoning, explanation, knowledge refinement and validation, and automatic classification of knowledge) is the goal of KBMS research.

Development Issues and Illustrative Projects

Commercially available KBMS technology is presently in its early stages of development. A variety of challenges must be understood and overcome before such systems will emerge from the laboratory. Several critical issues include performance, tradeoffs between flexibility and efficiency, the need for better requirements modelling languages, methodologies for knowledge acquisition, storage and knowledge capacity, and processing for large scale knowledge bases from both hardware and software perspectives. In this section, a summary of overarching KBMS research is given; specific

KBMS functionalities (e.g. expert systems for aiding user searches, natural language interfaces) are discussed in the **Advanced User Interface Capabilities** section.

KBMS and DBMS both employ the function of searching, although in distinctive ways. KBMS uses an inference search; DBMS, query evaluation. Inferencing is flexible both in terms of expression and in the addition of new knowledge with no impact on existing representations. DBMS query evaluation is much more rigid in its expressiveness and in the classes of data against which searches can be done. This approach is, however, computationally more efficient and can handle substantially greater amounts of data in a given storage capacity. Thus, in comparison to DBMS, the size of knowledge bases tends to be small; and the processing speed of searches, slow.

KBMS storage capacity is measured in objects (knowledge chunks composed of rules, frames, facts); current KBMS technology supports knowledge bases of about 10^3 objects. While there are KBMS capable of storing 10^6 objects under development [Brodie, 1986], SSP applications will require capacities which approach the range of present DBMS technology (or approximately 10^{10} objects). Also, KBMS typically are main memory intensive applications; unlike the DBMS, they presently do not rely upon secondary storage. Their adoption will drive the need for large random access memories. Further, increases in KBMS storage capacities will drive the need for better means of knowledge base maintenance, including knowledge verification, keeping the integrity of relationships between knowledge chunks in the face of updates, and index maintenance.

Knowledge processing performance is measured in LIPS (Logical Inferences Per Second) where 1 LIP is roughly equal to 100 to 1000 instructions per second per MIP of hardware. Current KBMS technology can support a rate of 10^2 LIPS per MIP over 10^3 object knowledge bases. In equivalent terms, DBMS technology now supports 10-30 LIPS per MIP over 10^{12} object data bases. Minimally, KBMSs should be capable of approaching the processing performance of DBMSs for large object data bases [Mylopoulos, 1986].

An ongoing KBMS research and development project is being sponsored by the Air Force and MIT's Sloan School of Management at the Knowledge Based Systems Laboratory, Texas A&M University. The goal of this effort is to provide technical support for a structured approach to strategic planning, tactical planning, requirements definition, design, construction, implementation, and maintenance of large scale distributed, heterogeneous, integrated information systems [Knowledge Based Systems Laboratory, 1987]. This work, while in its early stages, is of potential interest because the problems being addressed are comparable to those SSP would face in implementing KBMS strategies.

Another effort of interest is the CODOR project (Composite Document Effective/Extended/Expert Retrieval) being conducted at Virginia Tech University [Fox, 1986]. When completed, this system will combine advanced features for browsing; Boolean, p-norm, vector, and probabilistic retrieval; and knowledge representation models from artificial intelligence. The concept is to synthesize emerging capabilities from DBMS technology and the field of information retrieval, using as a physical architecture new storage devices such as CD-ROM. The user interface being developed draws on work from library science, linguistics, psychology, and computer science and may generalize to SSP applications.

Kimura [1986] has developed a prototype document processing system which builds on an object-oriented, abstract document model. This system has a sophisticated structure editor which is interactive and allows extensive windowing to see structural relationships among components. A software architecture which supports sophisticated translation tools for electronic manuscripts is also being developed [Mamrak et al, 1987]; this may solve some current problems with conversion among different representational formats. These and similar projects are evolving the components necessary for full-fledged KBMS functionalities.

Summary

An integration between current information technologies (such as DBMS, Computer-Aided Design, Text Management, and Image Management) and emerging artificial intelligence technologies (e.g. expert systems, natural language, and sophisticated knowledge representation schemes) is well underway. The result will ultimately be efficient management of large, shared knowledge bases. Initial KBMS are emerging in a variety of forms tightly or loosely coupled with DBMS technology, depending on intended use and the conceptual design principles of the developer.

Current implementations of KBMS fall short of the general demands of SSE documentation; however, a sufficient degree of capability does exist for the creation of prototype applications within the SSE environment. The ultimate capabilities of KBMS include:

- multiple knowledge representation schemes
- multiple knowledge directed applications
- context-free information and context-sensitive advice
- automated organization of knowledge

One result important to SSE is the potential emergence of new paradigms for software engineering through the development of KBMS technology.

Overall, the potential implications of KBMS research could be profound, as the next generation of information systems may become the first "knowledge medium:" humanity's conscious mechanism for tailoring its cognitive evolution [Stefik, 1986]. For example, in the final chapter of his book, The Selfish Gene [1976], Richard Dawkins suggests that ideas (he calls them "memes") are like genes. This opens up a myriad of analogies: meme pools, mimetic drift, mimetic mutation and displacement, recombinant memes. One spin-off of SSP work on advanced information systems could be new and powerful strategies which individuals and organizations could use to deliberately reshape their cognitive ecologies.

Alternative Knowledge Representation Approaches

The preceding section analyzed progress toward Knowledge Based Management Systems. This work depends in part on research into representational formats, for a fundamental aspect of any information system is its underlying knowledge representation. Symbols "re-present" reality, and the properties of any form of communication are shaped by the attributes of its symbolic substrate. For example, sometimes "a picture is worth a thousand words" because pictorial representational strengths are better suited to the needs of that particular knowledge transfer situation.

Historically, software documentation has been predominantly textual, consisting of annotations in the code itself coupled with descriptive written information. Such a knowledge representation approach is limited in several ways. As discussed earlier, descriptions based solely on text can be cumbersome compared to multi-media formats which allow easy incorporation of graphics, images, and animation. Text presents information in a sequential manner, but a person using software documentation frequently needs access to knowledge linked nonlinearly (for example, examining the detailed exposition for an annotation in the code requires jumping around in a text-based documentation system). Textual material is most easily searched for strings of symbols in close physical proximity (e.g. locating all paragraphs in a document which contain the phrases "software engineering" *and* "Prolog" *and* "expert systems"). Such key word retrieval systems may locate less than twenty percent of the references relevant to a particular search [Blair & Maron, 1985].

Since electronic media can support other formats than text, this section examines "hypermedia," an emerging knowledge representation approach, to determine its strengths and weaknesses for space station software documentation. First, hypermedia will be defined and its functions examined. Then, a brief overview of emerging hypermedia systems will be given. Finally, hypermedia will be evaluated as an alternative to linear multi-media formats for software documentation.

The Concept of Hypermedia

"Hypermedia" is a knowledge representation format composed of nodes of information connected by links. A "node" might contain text, data files, graphics, images, code, animation, or some mixture of these. Nodes can be arbitrarily large or small, but generally embody the equivalent of a few sentences to several pages of information; they can have multiple embedded links to other nodes.

"Links" are associative paths between nodes; they can be as simple as a connection between an origin and a destination or can have a variety of properties (such as purpose and direction). Links allow the creation of a conceptual web with complex interdependencies among nodes. This provides a framework for nonlinear representations of knowledge.

One way to conceptualize a simple hypermedia system is to compare this architecture of nodes and links to long-term human memory. Ideas are stored in the mind associationally; the word "apple" conjures up botanic, gustatory, computational, corporate, and theological dimensions. Imagine "apple" as a node, with multiple links spreading out to nodes describing these alternative meanings. Such a structure is one type of "semantic network" [Woods, 1975], in which links are determined by the relational meanings of the nodes' contents.

If an encyclopedia were organized on this basis, its knowledge would be more accessible than in the alphabetic sequential form currently used. Readers could browse by following links or could conduct searches which would include the link structure itself (e.g. find every paragraph which contains the words "apple" and "tree," but not "snake" *and* is linked to a paragraph containing the word "orchard"). Alternatively, if a visual representation of the link structure itself were available, one could navigate in the network of nodes by "flying" a route through the links or by "teleporting" from one node to another.

Shasha [1985] provides a more rigorous definition of such structures, describing hypertext as "text fragments embedded in a directed graph with labelled edges and instructions allowing users to traverse edges." He uses the concept of directed graphs in constructing a mathematically based fragment theory to optimize retrieval and presentation of knowledge. This work is directed toward developing a formal design strategy for electronic environments tailored to knowledge exploration.

Uses for Hypermedia

A nonlinear knowledge representation is superior to linear formats such as text for several types of applications. Researching a topic which is referenced in a variety of interrelated linear documents can be frustrating, since locating a part of the data required often provides little help for finding in other documents the remainder of the information needed. In contrast, a hypermedia system allows the user to follow a web of connections in tracing knowledge scattered in multiple sources. (A metaphorical description of this attribute would be the first needle found in the haystack acting as a magnet which then collects the remaining needles.)

This property of hypermedia also enhances the capability of documentation to serve as a source for training; a user unfamiliar with the knowledge space can be guided along prestructured paths. Currently, new workers often need substantial amounts of help from an expert human mentor to master the overall structure of a large system design project like the space station. A major productivity gain would be possible if a hypermedia documentation system could assume some of the responsibility for educating apprentices. In addition, the use of a nonlinear representation which mimics human associational memory may give all users greater recall of the material in the knowledge base, since the network's conceptual structure more closely mirrors their internal mental models of the information.

Another advantage of hypermedia is that using text as a method of organizing thoughts about programming, authoring, design, or problem solving can be difficult, since a linear format does not facilitate collecting and integrating a variety of approaches to the task. Current outline processors are limited to sequencing ideas hierarchically but, when concepts can be interrelated in a more complex manner, the brainstorming and synthesis aspects of knowledge creation are easier. Hypermedia as a representation enables both the capture of divergent mental models (as nodes) and their convergence into a coherent strategy (by linking these nodes into a semantic network).

Also, team collaboration on a project is empowered by a nonlinear medium because annotations and suggested revisions can be readily incorporated into a document. For example, a chunk of code and related documentation could be circulated electronically to multiple reviewers; the comments of each could be iteratively attached, labeled, and linked until the final hyperversion captured the collective wisdom of the group. In contrast, collecting from team members opinions and amendments on a document in standard text format is very unwieldy.

Finally, hypermedia formats support modularity of information, a vital strategy in large system development projects. The same node of information can be referenced from multiple locations, minimizing duplication and overlap. This allows greater customization of documents (through tailoring the order and availability of segments), reduces the volume of archival material (since information used in many documents is stored in a single place), and facilitates revision (because only one node need be altered to create changes throughout the knowledge base).

Thus, hypermedia is a representation system well suited for conceptual exploration, retrieval, training, retention, group collaboration, customization, and revision. The spectrum of functions a particular hypermedia system supports will depend on what balance among these activities is most important to users.

Range of Features

Describing hypermedia applications in terms of the operations they enable can be quite complex. Conklin [1986a] delineates a taxonomy of typical features which includes:

- support for hierarchical or non-hierarchical link structures

- allowing multiple versions of nodes or links
- support for paths (many links connected into a single permanent object)
- availability of functions which facilitate searches through the hyperdocument
- support for multiple concurrent users
- availability of sophisticated editing tools
- "multiple parenthood" assignments (for inheritance of attributes)

Many of these characteristics would be essential for a space station knowledge base.

As "second generation" hypermedia systems evolve, the range of features unique to nonlinear representational approaches may expand considerably. For example, in current systems nodes and links are objects within hypermedia; a next step would be the inclusion of higher order representational structures (e.g. subnetworks) as objects to be manipulated. Such "composite nodes" are already partially implemented in some hypertext systems (for example, NoteCards, which is described below).

Also, supporting different types of nodes is an important capability for hypermedia systems, since part of their power is that a correspondence can be created between objects in the world and nodes in the hypermedia knowledge base. By using a variety of node types, the user can model a real world phenomenon in a flexible network. Templates within typed nodes enhance the completeness of information entered and may aid in computational inference on node contents. (A section which follows shortly, "A Software Documentation Hypertext," illustrates the application of these general precepts in a real world programming environment.)

Similarly, the availability of different link types allows the knowledge creator to move beyond a linear flow of information (in which concepts are either included or discarded) to a complex representation that gives the knowledge consumer a web of choices. The Intermedia system (described below in the section on "Hypermedia Systems for Presenting Information") illustrates the power of typed links in customizing the user interface.

Focusing on nodes leads to analysis or reductionism, in which a complex phenomenon is understood by dividing it into smaller parts. In contrast, a link orientation promotes synthesis or systemic thinking, where the processes which connect components of an entity are the dominant determinants of its behavior. Hypermedia as a representation supports both types of cognition; a structured network is analogous to an optical illusion, in which different images can be seen depending on which gestalt the observer superimposes on the information presented. This support for multiple mental models (recontextualization) is a strength of a nonlinear format.

Current Systems

Work in hypermedia is in an exploratory stage, and the systems currently in use are prototypes or are in the early phases of usage. These systems are generally described as hypertext to indicate that they do not yet support the full range of representations implied by hypermedia. Robust second generation projects which build on the lessons from these initial attempts will emerge over the next several years. A few hypertext applications which illustrate features of particular interest for space station documentation and the SSE are summarized below.

A Software Documentation Hypertext

Dr. Walt Scacchi at the University of Southern California (USC) has developed a hypertext electronic encyclopedia which organizes documentation as part of a larger System Factory software development methodology [Scacchi, 1987]. The eight life cycle stages of the System Factory approach, which utilizes C as its programming language, are Requirements Specifications, Functional Specifications, Architectural Design, Detailed Design, Coding, Testing, Integration, and Usage and Maintenance.

The Document Integration Facility (DIF) maintains a document for each life cycle stage of each project in the System Factory. Its hypertext architecture links nodes into sections which form these documents. A sequence of Basic Templates (BTs) are used to organize the networks of nodes; BTs are structured to facilitate search, versioning, revision control, and attribution [Garg, Jazzar, & Scacchi, 1986a].

For each section of the documentation, the hypertext encyclopedia builds up a database which contains 1) key words describing its contents and 2) the type of information these nodes contain (natural language, functional specifications, program code, design specifications, relocatable or executable object code for a module, bit mapped displays). Two modes of use are available which customize the interface to managerial or engineering applications. DIF is designed to:

- improve completeness and consistency;
- customize documentation for different types of users;
- provide greater traceability of development decisions;
- enhance the integration of text, graphics, and code;
- reduce the burden of repetitive information;
- encourage team oriented development; and
- enable better management of revisions.

This hypertext documentation system is integrated into the System Factory as part of a larger set of tools, which include a specifications analyzer and simulator, a module interconnection and interface definition processor, a language-directed editor, and a series of UNIX operating system functions. DIF is structured as the central resource which provides a uniform interface to the other tools. The System Factory and DIF are evolving in their sophistication and generalizability as Scacchi's work progresses.

Such a representational architecture has considerable advantages for software engineering. A frequent source of problems in software development and maintenance is that changes are made to one part of the program without an understanding of those alterations' implications for the remainder of the code and documentation. Systems like DIF can use a truth maintenance system (or some other form of inference engine) on top of a hypermedia format to propagate the side effects of any change to the program. As a result, programmers can determine all the direct and indirect consequences of a proposed alteration [Narayanaswamy & Scacchi, 1987].

Another major advantage is that potential processing bottlenecks among key software modules can be identified through examination of the hyperstructure; this is particularly useful in large scale systems which are configured over time. In addition, a template-based format facilitates capturing the rationale for design decisions, and the modular format of a node structure may enhance code reusability.

Hypertexts for Computer Aided Design (CAD)

The Tektronix Neptune System has been developed as an open, layered architecture which has two components: a transaction-based server called the Hypertext Abstract Machine (HAM) and a graphic interface written in Smalltalk-80 [Delisle & Schwartz, 1986a]. The goal underlying the development of Neptune is to create an electronic documentation system for all information pertaining to technical projects (including requirements specifications, designs, implementations, evaluations, and tests). Such a system could become the heart of a design support environment tailored to facilitate multi-person cooperation in large-scale software development. Tektronix also plans to use HAM as an underlying architecture for their computer-aided software engineering (CASE) technology.

Traditional databases give inadequate support to version control and configuration management in computer aided design; another weakness is that hierarchical and relational models do not map well to the types of data stored in CAD systems [Penedo & Stuckle, 1985]. A nonlinear representation's more flexible architecture can solve many of these problems, and application layers can be built on top of a generic hypertext system such as HAM. For CAD, such an application layer could include VLSI design tools, high level language compilers, or document processors.

HAM implements a number of important capabilities: version histories, traversal and query mechanisms, "demons" (which invoke code when a specific event occurs), multi-user editing, transaction-based crash recovery, and a generic architecture. Another useful attribute is a general mechanism for attaching semantic information (e.g. attribute/value pairs) to nodes; this allows higher level applications to define customized accessing mechanisms to the hyperdocument. Neptune, which is designed for software engineering, provides a variety of specialized browsers, hierarchies of documentation materials, and sub-graphs for attribute-based access.

Since authors may need to modify the same information simultaneously, partitioning a shared hypertext database--with provisions to join the partitions at controlled intervals--is a vital feature for large system development projects. A difficult technical challenge for any hypermedia system is partitioning across gateways and local area nets in a distributed processing environment. Neptune incorporates an approach called "contexts" (working collections of nodes and links), which seems promising in attacking the partitioning problem.

A second hypertext system for software design, the Personal Information Environment (PIE), has been implemented in an experimental version [Goldstein & Bobrow, 1984]. The goal of PIE is to provide the designer with various perspectives (the performance view, the reliability view, the security view, the management view) on an evolving software system. Each node is implemented with multiple embedded perspectives; constraint propagation and contextual mapping are also pieces of PIE.

Hypertexts for Exploring Ideas

NoteCards, a hypertext system developed at Xerox Palo Alto Research Center (PARC), is an extensible environment for manipulating ideas. "Idea processing" consists of three types of activities: acquisition, analysis, and exposition [Halasz, Moran, & Trigg, 1987]. NoteCards is designed to:

- facilitate the manipulation of symbols that represent ideas and their interconnections,
- store and retrieve structured networks of ideas, and
- provide tools for tailoring the generic hypertext representation to the needs of specialized domains.

The metaphor underlying this hypertext system is a set of 3x5 paper note cards, electronically represented as nodes. Links interconnect these nodes into structured networks, through which the user can navigate or search. Customized types of nodes and links can be created. For example, Browser cards facilitate a user's traversing or editing subnetworks (the equivalent of composite nodes). FileBoxes are specialized cards which provide a hierarchical organization for cards independent of their network interconnections.

Typical applications of NoteCards include authoring papers, providing a medium for competitive argumentation among different hypotheses, and supporting a computer-based training environment. As the range of these uses suggests, this hypertext is designed to be very flexible within the overall concept of "idea bases" (as opposed to databases, management information systems, or knowledge bases). As discussed earlier under Uses for Hypermedia, nonlinear representations are well suited to the divergent/convergent thinking central to idea processing. (One of the NoteCard's researchers also developed TextNet, an on-line semantic network for organized access to technical materials [Trigg & Weiser, 1986].)

A crucial requirement of large system development projects is that participants' work be mutually intelligible, even when they have no contact other than through electronic communication. Collaboration involves activities that deal with 1) the substance of the work, 2) annotations about the work, and 3) procedural discussion about processes which support task performance. Hypertext as a representation medium is excellent at supporting such a collection of nested, referential communications. Current research at PARC is exploring how NoteCards can be tailored to facilitate computer-supported collaborative work (such as draft-passing for a shared project notebook, simultaneous authoring on a single notefile, or guided tours for presenting ideas to others) [Trigg, Suchman, & Halasz, 1986].

A Hypertext Programming Environment

Boxer is a programming language with nonlinear linking capabilities [diSessa, 1986a]. The goal underlying Boxer's development is to produce an environment for programming which is powerful, but easy to learn and use (even for novices or children). Human factors and cognitive science approaches are central to the design of this programming system, with an emphasis on advanced features for presentation, structuring, and contextual application. For example, Boxer is well suited to direct manipulation of standard programming objects, device programming, constructing customized interfaces, and building computational "microworlds" (limited alternative realities)--a concept which will be discussed in detail in the **Advanced User Interface Capabilities** section of this study.

The fundamental unit of information in this programming environment is a "box" (similar to a node that is customized for code manipulation). Boxes can be hierarchically nested; for example, a Boxer program is a box which contains different types of internal boxes. These include boxes for input and output variables and other boxes which determine how variables are processed. The hypertext aspect of Boxer comes from its inclusion of "ports," which allow boxes to be viewed and altered at multiple places in the overall programming environment. Ports have capabilities for link types, paths, procedural attachment, and node attributes, so the same piece of code can simultaneously be part of many programs and accessible from each.

Boxer's integrated functionalities include text processing and structured filing, the ability to use and modify prewritten programs, database features, graphics capabilities, and tools for programming from scratch [diSessa, 1985]. This environment illustrates some promising directions for improving software development and demonstrates how supporting nonlinear networks within code itself can simplify the programming process.

Hypermedia Systems for Knowledge Transfer

Several projects have focused on using a hypermedia format to increase user mastery of the knowledge in a database. One such program, Intermedia, is being developed by the Institute for Research in Information and Scholarship (IRIS) at Brown University. This hypermedia project is the latest in an evolutionary series of electronic documentation systems [Yankelovitch, Meyrowitz, & van Dam, 1985].

Intermedia enables browsing through multi-media information which is linked, cross-referenced, and annotated. System authors can access an integrated set of tools which include a text processor, a graphics editor, a timeline editor, a scanned-image viewer, and an application to view and manipulate three-dimensional models. Future plans include the development of segment, map, video, and animation editors, as well as access to CD-ROM data [Garrett, Smith, & Meyrowitz, 1986].

This hypermedia system has been designed using object oriented programming to facilitate building the web structure [Meyrowitz, 1986]. Maps which present visual representations of the network are used to guide viewers through the system. Current research at IRIS is targeted to the development of sophisticated navigational systems, customizing the user interface, versioning, and "hot links" for propagating information through the web. Intermedia materials are used as a resource in several courses on the Brown campus, so the system is designed to handle a multi-user environment of workstations connected by local area networks.

A second group of hypertext knowledge transfer applications have been developed using HyperTIES (Hypertext based on The Interactive Encyclopedia System) at the Human-Computer Interaction Laboratory, University of Maryland. TIES was developed to facilitate user browsing through instructional databases with an underlying hypertext architecture. The system is designed to maximize ease of use for both readers and authors and has an explicit instructional model [Shneiderman & Morariu, 1986]. HyperTIES is being expanded to include an advanced browser (with string search, bookmarks, multiple windows, and user annotation) and sophisticated authoring capabilities. Videodisc and touchscreen support are also under development [Shneiderman, 1987a].

The Human-Computer Interaction Laboratory has conducted a series of rigorous empirical studies of various design alternatives for the user interface (e.g. mouse versus arrow keys, embedded versus explicit menus, alternative screen sizes). Since repeated research has shown that reading text from screens is thirty percent slower than from typewritten paper documents [Shneiderman, 1987b; Gould et al, 1987], such productivity studies are vital to the evolution of electronic documentation.

A prototype electronic encyclopedia with a hypermedia substrate was partially developed at Atari [Weyer & Borning, 1985]. The four metaphors used to guide design of the knowledge base were a *model* (a representation of some knowledge), a *tour* (a particular path through some model), a *filter* (an intelligent interface which tailors a model to a particular user), and a *guide* (an intelligent electronic agent that selects tours and provides help). Object-oriented hypermedia links and embedded simulations provided a richer representational structure than the typical linear encyclopedia format.

Another knowledge transfer hypertext project is Xanadu, an on-line proprietary system developed by Ted Nelson (who coined the terms "hypertext" and "hypermedia" in the 1960s). Xanadu is designed to be generic as a hypertext application so that many different types of uses can be supported. The fundamental metaphor Nelson espouses is a unified, global literary environment [Nelson, 1987].

The fundamental node unit in Xanadu is the document, which may have windows (links) to any other document in the knowledge base. Versioning is supported, and sophisticated organizational and indexing strategies are under development. Nelson is devising a strategy for electronic accounting and distribution of royalties, since copyright issues are a major concern for electronic storage media.

An optical disc research prototype, Palenque, is being developed at the Bank Street College of Education [Wilson, 1987]. This knowledge transfer hypermedia system uses a CD-ROM (Compact Disc-Read Only Memory) with General Electric's Digital Video Interactive (DVI) technology. Palenque is designed so that children and their families can browse a database about a Mayan temple; the information is organized both topically (as a museum with four theme rooms) and spatially (so that the user can "walk" through the temple). When supplemented by a structured network of graphics, sound, text, narration, slides, motion video, and simulation, "virtual travel" is a powerful motivational strategy, as prior work with spatial data management attests [Lippman, 1984; Bolt, 1979; Mohl, 1982].

As an example of an application specialized to software development issues, a Computer Based Ada Training System (CBATS) with a hypertext format has recently been developed by Softech for the Pacific Missile Test Center in Point Mugu, California [Larson & Rienzo, 1987]. CBATS utilizes existing Ada documentation and training materials (e.g. Practitioner's Guide to Ada, Army Ada Training Curriculum) reconfigured into linked chunks which illustrate interrelationships. This system illustrates the strengths and limits of the rapid, low cost transfer of conventional information to a nonlinear format.

KMS™ (Knowledge Management System) is a commercial hypermedia framework which grew out of work on the ZOG system (a Navy-sponsored, first generation hypertext developed at Carnegie-Mellon University from 1972-1985). KMS is designed for an architecture of networked heterogeneous workstations and requires power equivalent to a Sun or Apollo computer. This hypermedia system provides a framework for organization-wide collaboration on applications such as electronic publishing, software engineering, project management, computer-aided design, and on-line documentation [Akscyn, McCracken, & Yoder, 1987].

A "second generation" hypermedia system, HyperActivity, is currently being designed at the Microelectronics and Computer Technology Consortium (MCC). A precursor to HyperActivity was PlaneText, a proprietary Unix-based hypertext application [Gullichsen et al, 1986]. Also, an advanced three-dimensional graphics interface to semantic networks (SemNet) has been developed [Fairchild, Poltrock, & Furnas, 1987].

Frank Halasz, one of the architects of NoteCards, indicates that the HyperActivity project will implement the following features:

- composite, multi-media nodes
- advanced search and query (using attributes and context)
- versions and histories
- sophisticated navigational displays
- support for collaborative work (e.g. contribution tracking)
- interface tailorability
- support for embedded inference engines

Such work on second generation hypermedia systems is important because a space station electronic documentation system would require many of these features [Halasz, 1987].

As the systems described above illustrate, hypermedia is a powerful approach for knowledge transfer. The strengths of this representation are 1) immediate access to information scattered among various sources and 2) guided paths which sequence users' exposure to data and make interrelationships explicit. Cognitive principles of learning which support the concept of hypermedia-based educational environments include active structural networks, schema theory, web learning, and generative processing [Jonassen, 1986].

Other Hypermedia Systems

This brief summary of illustrative research in nonlinear knowledge representations has only scratched the surface, as coverage was limited to major projects which have special applicability to space station and SSE documentation needs. (A detailed and technical overview of current hypertext systems is given in Conklin [1986a]; summary for a more general audiences, in Conklin [1987] and Pea [1987].) Commercial interest in hypermedia is growing rapidly; four nonlinear applications have appeared for Apple's Macintosh™ (one bundled as system software) and two for IBM PC™ compatibles. Over the next several years, hypermedia programs may proliferate as rapidly as expert system shells!

Operational Challenges

Limits typical of first generation systems (e.g. computational delays, deficiencies in visual representation of the web structure) will disappear over the next several years as second generation hypermedia systems and more powerful workstations evolve. Deeper problems that may constrain the ultimate usefulness of hypermedia are disorientation, cognitive overhead, combinatorial explosion, and collective communications dysfunctions.

Information which is organized in a complex manner poses a potential problem of user disorientation. In a linear medium, one can readily evaluate the extent to which a document's information has been traversed (how many pages read, how many left) and where a particular piece of data is located (chapter, section, paragraph). Large hyperdocuments may be more confusing. In a web of thousands--or millions--of nodes, how does one define a location in the network, establish a desired direction to move, or blaze a trail indicating those nodes already scanned? In a non-hierarchical structure, what type of system should be used to indicate where a piece of data has been stored? Research is underway to develop sophisticated visual-spatial interfaces to address these problems [Fairchild, 1984], but this problem is very challenging for networks with large numbers of nodes and links.

Even if a user familiar with a particular network experiences no disorientation, working in a hypermedia knowledge base entails some extra cognitive overhead. If entering material, the author must think carefully about how to link the information being added to the web which already exists. At each node they encounter, users must choose which link to follow from multiple alternatives and must keep track of their orientation in a complex multidimensional structure. The richness of a nonlinear representation carries a risk of potential intellectual indigestion, loss of goal directedness, and cognitive entropy. Unless a hypermedia system is designed carefully from a human factors perspective, increasing the size of the knowledge base may carry a cost of decreasing its usability. Lenat's work [1986] on building a large knowledge base of real world facts and heuristics with embedded reasoning methods is developing a theoretical framework for resolving these types of problems.

The availability of multiple types of representations in a hypermedia system compounds this problem. Access to representational alternatives allows users to tailor input to their individual cognitive styles and enables authors to choose a format well suited to the material being entered. However, coping with multiple formats adds to cognitive overload, and little is known about which representational ecologies are functional for different task situations.

"Combinatorial explosion" (being overwhelmed by alternative possibilities) is another type of potential limit. Consider the two extremes of interconnections among nodes: if all nodes are linked to each other, the network is meaningless; if each node is connected to only a preceding and a following node, the medium is linear. Hypermedia formats are useful only when balanced between these extremes.

The number of links generated by adding an additional node to a network will vary depending on the type of knowledge being stored, the objectives of the documentation, and the sophistication of the user population. Suppose that many vital, subtle interrelationships exist in the network's material. Although some new nodes will simply annotate single existing nodes, a substantial proportion of nodes that are added may require multiple links. The larger the knowledge base, the more links will be needed to integrate the additional material (because more nodes which discuss related issues will already be present). In a worst case situation, a combinatorial explosion would take place, with each new node adding so many links that the difficulty of comprehending and maintaining the web exceeds the benefits which a nonlinear format provides.

One strategy for solving this problem is to aggregate subnetworks into composite nodes which chunk material on a higher level of abstraction. Such an approach is being explored in second generation hypermedia systems--but creating another dimension of hierarchy complicates the representational architecture and, unless implemented in a manner transparent to users, may increase disorientation and cognitive overload.

These potential limits are particularly acute in on-line, shared hypermedia systems. The user of a collegial electronic knowledge base may find that, since last entering the system, familiar paths have changed and new material has appeared. Links that seem intuitively obvious to the author adding them may be puzzling to others. "Tower of Babel" communication dysfunctions are possible with a large knowledge base in which multiple users can alter the fundamental medium of interaction.

Possible problems of disorientation, cognitive overhead, combinatorial explosion, and collective communications dysfunctions reflect the intricacy of working with a knowledge base rather than a database. Knowledge is intrinsically complex and, as discussed in this study's **Introduction**, transforming information to knowledge involves gaining a goal-directed, contextual understanding of the application domain. Utilizing an underlying representation based on hypermedia will require more sophisticated skills--a new type of "literacy"--from personnel in every role associated with space station documentation. However, knowledge is much more powerful in task performance than data or information, and the effort required to use a knowledge base for documentation may well be outweighed by gains in efficiency and effectiveness at every level of project operations.

Evaluating the Usefulness of Hypermedia

How do hypermedia and linear multi-media formats differ in their usefulness for knowledge representation, particularly software documentation? One approach to evaluating the strengths and weaknesses of each is to examine the underlying properties of these different formats. Any mechanically embodied intelligent knowledge representation system must have three elements [Brachman & Levesque, 1985].

First, a well-specified language is necessary to express the knowledge. This is not a trivial requirement, for the underlying properties of this language can be very influential in shaping the performance of the knowledge base. As an illustration, anthropologist Edward Hall [1973] describes how some American Indian languages treat colors as verbs (e.g. the sky *blues*), whereas in English colors are adjectives (the sky is *blue*). Modern physics has determined that colors sometimes behave as if they were verbs, other times as adjectives (since light has both wave-like and particle-like properties). Advanced optical concepts can be expressed--and perhaps discovered--more easily in languages where colors can be both active and passive attributes. Optimally, a knowledge representation language is tailored to the specific domain for which it will be used.

Second, a representation system must provide an inference mechanism for revealing to the user facts implicit in the knowledge. For example, if the knowledge base includes the propositions "Socrates is a man" and "all men are mortal," then the implicit fact "Socrates is mortal" should be readily accessible. Researchers in knowledge based systems are exploring alternative representational structures (i.e. rules, frames and schema, logic) which enable such computational inferencing.

Third, work in artificial intelligence has determined that specific knowledge in a domain is very important to task performance ("in the knowledge lies the power"). As a case in point, consider the design of an expert system for software programming. One approach would be to construct a system which understood the general theory of Turing machines and had a powerful inferencing mechanism. This would obviously be inefficient compared to building a system with detailed knowledge about algorithms and language primitives likely to be useful. It is better to know than to be able to figure out [Szolovits, 1986]. The four major bottlenecks in scaling up to the system sizes requisite for space station applications are knowledge acquisition, knowledge consistency and completeness, large knowledge base manipulation, and interface technology [Friedland, 1987].

Given this framework for evaluation, hypermedia is theoretically superior to linear multi-media formats in several ways. The representation languages that hypermedia can support are potentially more expressive because nonlinear semantics (composite nodes) and syntax (attribute/value pairs, neighboring link structure, node and link types) are richer. Hypermedia formats enable easier computational inference mechanisms because truth maintenance approaches can be used to propagate implications through the structured network. As discussed earlier, knowledge is contextual, goal-directed, and involves multiple perspectives; hypermedia provides a regional semantic environment for a node, as well as supporting alternative mental models (analysis, synthesis).

Whether this potential is outweighed by the disadvantages of nonlinear representations is a more complex question. Much depends on the type of material in the database and the goals of the person involved in knowledge creation, capture, transfer, or utilization. For example, Shasha [1986] contrasts the activity of reading a short story collection with that of consulting a guidebook. A nonlinear format would add little to the compendium of stories, which are intrinsically "chunked" and are not extensively cross-referenced. However, a hypertext guidebook that would allow searching for restaurants by location, price, or type of food would be most useful. In general, databases are good candidates for nonlinear representation if they contain many pieces of information, each with multiple attributes, and if users need to see different groupings.

As discussed earlier, a hypermedia format can be useful in situations involving conceptual exploration, retrieval, training, retention, collaboration, customization, and revision. The SSE (and space station documentation systems in general) meets the criteria above of many chunks of multi-attribute knowledge needed by different types of users. A nonlinear knowledge base which minimized problems of disorientation, cognitive overhead, combinatorial explosion, and collective communication dysfunctions could be very useful.

Other Potential Representation Formats

A variety of alternative formats for representing knowledge are being investigated by artificial intelligence researchers. In designing a documentation system, choosing among these representations involves considering many aspects of a knowledge transfer medium. For example, Wenger [1987] has developed a theory of machine-mediated knowledge communication. His important dimensions for the design of a knowledge base include:

- the breadth of information a documentation system incorporates;
- the depth to which each topic is treated (including whether the knowledge base is structured to correct common user misconceptions);
- the "grain size" of the material;
- the categories used for epistemological organization;
- the inclusion of multiple alternative perspectives and user tailorability; and
- the computational cognitive architecture, including its internal and external representation formats.

Clearly, how knowledge is represented can affect its transferability.

Wenger argues that a domain and a population of users together define a set of possible states differentiated by the extent, correctness, organization, and viewpoint of each person's knowledge. The goal of a documentation system would then be to move users through a sequence of increasingly sophisticated "knowledge states" to optimize task performance. Such a viewpoint illustrates the complexity underlying the choice of a knowledge representation.

In addition, the properties of the conceptual substrate selected will have implications for the types of computational inference (including commonsense reasoning) that an intelligent documentation system could support [Szolovits, 1985]. Choosing an inflexible, limited representation could preclude later enhancement of the documentation system with automated inference and reasoning techniques. Fortunately, hypermedia seems a versatile substrate on which a wide range of knowledge representation formats can be overlaid [Marshall, 1987].

Beyond linear multi-media formats and associational representations similar to hypermedia, possible alternatives which empower machine-based intelligence include structured object representations (frames and schema), procedural representations (e.g. FORTRAN code), formal logic-based representations, and a variety of emerging approaches (i.e. default, analogical, probabilistic, qualitative, evidential, diagrammatic, hybrid) [Brachman & Levesque, 1985]. As the field of knowledge based reasoning matures, additional types of representations are likely to appear.

Summary

Developing a protocol for selecting a knowledge representation is beyond the scope of this study. A variety of viewpoints about the design of knowledge base management systems is presented in Brodie & Mylopoulos [1986], but no consensus emerges on this issue. A conservative approach would dictate:

- selecting a documentation architecture which allows maximum flexibility (the capability of utilizing all these types of representations),
- focusing current efforts on developing integrated linear formats (text, graphics, images, code), and
- devoting significant resources to exploring hypermedia as an simultaneously supported alternative representation.

The central problem is not to design space station documentation systems which are optimal from the knowledge representation point of view (though that is difficult enough). The real challenge is creating a joint human-machine knowledge base capable of accomplishing this large system development project. Evaluating alternative knowledge representation approaches from this perspective involves partitioning task performance into machine-supported and machine-controlled domains and determining the role an electronic documentation system should play in each. The next two sections of this study **(Advanced User Interface Capabilities, Computer-Supported Cooperative Work)** analyze this issue.

Advanced User Interface Capabilities

A crucial aspect of any application is its interface, since the utility of any computer-based tool can be no greater than its accessibility. If mastering a job performance aid is perceived as too complex, many workers will ignore its existence, and most of its users will not tap the full productivity gains it enables. Unfortunately, the more powerful an application, the more complex its interface must necessarily be to support the range of features available. This poses a problem for large scale documentation systems, such as the SSE, which incorporate a wide variety of different types of functionalities.

This section presents some design "heuristics" (rules of thumb) which may be valuable in constructing the user interface for space station documentation systems (and for KBMS in general). This set of issues is not covered in much detail because human factors considerations are already an important aspect of SSE development. NASA personnel, contractors, and grantees are sophisticated in human-computer interaction, as indicated by the research programs of Campbell, Nagel, Nickerson, Rudisill, NASA's Ames Human Factors Research Division, Shneiderman, and Tullis. A brief discussion of this topic is merited, however, since user-centered system design is a rapidly advancing field and the interface of advanced documentation systems may evolve considerably over the next decade.

Goals of User Interface Design

As researchers gain experience in constructing systems for knowledge creation, capture, transfer, and utilization, themes emerging as important design objectives include:

- building resilient interfaces which facilitate management of trouble, but are not "idiot-proof;"
- developing user interface management systems (UIMS) which offer intelligent assistance to novice or occasional users, but also incorporate powerful command sets for sophisticated practitioners; and
- providing support which aids the user in making decisions, but does not recommend solutions.

Cognitive engineering approaches based on human factors, cognitive science, and ergonomics are being developed to implement these heuristics [Norman, 1987].

The complexities of human interaction with an advanced computational environment preclude anticipating all possible problems and misunderstandings, so the goal of an "idiot-proof" interface is a fallacy. Instead of attempting to avoid all possible problems for any user (a strategy likely to render the power of a knowledge base inaccessible), the focus of design is shifting to managing trouble and providing resiliency.

The concept of "repair" (supporting humans in recognizing and correcting difficulties) is becoming central to advanced interface construction [Brown, 1986].

A fundamental precept underlying this approach is human control, which requires that the user have a common-sense understanding of system functioning. The interface needs to be a "glass box" which makes operations sufficiently transparent that people can apply them in the face of malfunctions, new situations, and errors. Such user control also allows tailoring the system interface to a person's level of expertise, individual cognitive style, and role.

If instead the system competes for control of the decision making process by offering solutions rather than advice, then problems can arise. For example, situations in which two parties have overlapping authority and responsibility can easily lead to double-bind dynamics. Also, the less than optimal performance of current expert systems necessitates human editing of machine choices to filter out poor selections; this can undercut many of the productivity gains of an automated knowledge based system [Woods, 1986].

Complex systems such as the SSE have a variety of attributes which make them hard for users to understand and control. Unlike mechanical artifacts, information systems are opaque: function cannot be inferred from structure. As an illustration, by examining the parts of a lawnmower and their connections, users can form a qualitative causal model of how it operates. With physical systems, people can develop mental models based on real world experiences with wheels, springs, screws, etc [Gentner & Stevens, 1983]. A comparable analysis is not possible with a word processor. At present, beyond simplistic analogies to file cabinets, causal metaphors which provide semantic rationalizations for information systems do not exist.

This lack of transparency in the functioning of documentation systems is heightened by their ability to support multiple processes simultaneously, so that a person is confronted by several interacting "black boxes." A related difficulty in understanding information systems is their functional complexity. Users encounter many linked applications (word processors, programming tools, electronic message systems), each of which has hierarchical layers of features.

A final level of uncertainty is added by the ambiguity inherent in any interactive system. Just as understanding a person's utterances often draws on a shared comprehension of the subject under discussion, collaborative action between a user and an intelligent tool requires a mutual sense of purpose. Written instructions for assembling an artifact (such as a bicycle) sometimes illustrate this problem. The steps to follow are confusing when encountered by a novice but, when all the parts have been connected, the directions retrospectively make sense. A misunderstanding about the "conversational postulates" underlying an interaction with an information system is the cause of many operational problems [Suchman, 1985].

These five attributes of information systems--opacity, a lack of causal metaphors, multiple simultaneous processes, functional complexity, and ambiguity--make interface design for human understanding and control a difficult objective to achieve [Brown, 1986]. However, approaches which may resolve these problems are beginning to emerge, driven by a recognition that user acceptance and mastery of advanced job performance aids is central to increasing productivity.

Illustrative Promising Directions

One limit on user control of a documentation tool is that the person needs a mental model of system operation so that unexpected troubles in executing tasks can be repaired. Research on how people use mental models to understand complex systems indicates that many fundamental questions need empirical study [Rouse & Morris, 1986].

Semantic Rationalization Strategies

A good conceptual model of an interface is clear, covers the full range of system functioning, and uses a metaphor rich enough to capture the spectrum of available features. For example, thinking of a word processor as "like a typewriter" is limiting because features such as cutting and pasting text, searching for strings of characters, or justifying paragraphs are functions omitted by this metaphor [Miller, 1986].

Self-explicating systems are an important emerging design strategy. One type of self-explicating interface uses graphics to create objects that are manipulable by the user in the same manner as their real world counterparts. For example, a doorknob icon induces an impulse to turn and an expectation that something will open. Such a "mimetic" interface is an imitation of the semantics of the underlying real-world referents [Laurel, 1986]. The system designer can build on universal common-sense knowledge in the human community without having to impart new skills when a person masters operational commands.

"First-person" (or "direct manipulation") interfaces allow users to work directly with the domain (e.g. turning a doorknob icon). [In contrast, "second-person" interfaces (discussed later) require the user to give syntax-laden commands to a second party, the computer, which carries out the actions.] Direct manipulation interfaces provide [Shneiderman, 1983]:

1) a continuous representation of the object of interest,

2) physical actions or labeled button presses instead of complex syntax, and

3) rapid incremental reversible operations whose impact on the object of interest is immediately visible.

Examples include graphical programming environments such as ThingLab [Borning, 1981], computer-aided design (CAD) interfaces, the Macintosh operating system, and spreadsheets.

Direct manipulation interfaces are limited in their ability to execute repetitive operations, handle variables, distinguish individual elements from a class of similar elements, and control actions with precision [Hutchins, Hollan, & Norman, 1987]. Nonetheless, first-person interfaces are attractive because they enhance user understanding and control by supporting common-sense, mimetic operations. For example, the Whiteboards system is the electronic equivalent of an office corkboard; users rapidly master elements of Xerox's Cedar programming environment by interacting with icons in a documentation database as one would manipulate equivalent collections of objects on real-world corkboards and whiteboards [Donahue & Widom, 1986].

An ultimate outcome of work on mimetic interfaces may be the construction of full-fledged artificial realities. These interfaces would utilize 3-D imagery, elaborate user input devices (such as gesture gloves for grasping computational objects), and hypermedia representations to create an alternative universe controlled by the user and optimized to performing a particular set of tasks [Fairchild & Gullichsen, 1986]. Gibson [1984] describes a fictional, but plausible society in which databases are accessed through a collective artificial reality.

The three components of an artificial reality are imagery which immerses the user in a visual space; modeling which allows the images presented to behave in consistent, meaning ways; and an interface which allows interaction with the images in a manner similar to real world experiences [Foley, 1987]. For example, the head-mounted display developed by Fisher, McGreevy, and Humphries at NASA's Ames Research Center projects to its wearer the images that a robot is "seeing." Such a display could be augmented with a device (such as the DataGlove™) that would allow the user to "grasp" these images and manipulate them, receiving pressure feedback from the glove comparable to the sensory stimulation that would be received from the object being modeled. A DataSuit to cover the human frame might be the next step.

"Microworlds" (interactive simulated environments with mutable rules of operation) provide limited alternative realities tailored to education. For example, the Alternate Reality Kit (ARK) allows users to manipulate the laws of physics and experience how everyday activities (such as throwing a ball) change [Smith, 1987].

Object-oriented interfaces of this type are powerful ways to increase motivation and understanding.

Educating users on how an information system functions is important because otherwise interactions are "magical." People can begin to use a non-mimetic, second-person interface by memorizing nonsensical (to them) commands which accomplish specific system actions. However, as tasks become more complex, the number of incantations to be remembered grows rapidly and may become overwhelming. Further, when trouble arises, users have no idea how to remedy the situation unless they previously have memorized a magical way of responding to this particular predicament. Because of these problems, training for space operations emphasizes a deep conceptual understanding of command sequences [Dede, 1987a]. Similarly, space-related documentation systems necessitate non-magical interfaces, and direct manipulation strategies are one promising direction.

Coordinating Multiple, Complex, Simultaneous Processes

A second limit on user control of a documentation tool is that many applications may be active at the same time, each having hierarchical layers of features. Such functional complexity requires a mastery of procedural skills that many users find difficult to attain [Sheil, 1982]. Several approaches to interface design are addressing this problem.

One strategy is to support user "linearizing:" performing several tasks while switching among them [Cypher, 1987]. Humans are adept at linearizing in everyday settings; for example, most people can walk, chew gum, and converse with someone simultaneously. However, multiple parallel activities carry a cognitive overhead: scheduling among competing priorities can become complex, the context of an interrupted activity must be stored for later recall, and actions must be sequenced for mutual compatibility. For example, crossing a busy street may suspend conversation temporarily, and enunciating a word may require shifting the wad of gum.

Windows are one approach to linearizing computational activities; they create multiple virtual screens, each with its own activity context. Card and Henderson's ROOMS [1987] is a window manager system with a mimetic interface. Each room (window) has furniture (computational tools appropriate to a particular task) and doors (traversals to related applications). Such a strategy is a promising alternative to interfaces which impose a single integrated structure on a package of applications (e.g. Lotus' 1-2-3™).

The Goals, Operators, Methods, and Selection Rules Model (GOMS) [Card, Moran, & Newell, 1983] provides a theory for how users organize functional complexity.

Domain goals are accomplished by selecting the most effective strategy from alternative sequences of system actions. Two human barriers to attaining this type of expertise in managing complexity are users' production bias and assimilation bias [Carroll & Rosson, 1987]. People want to accomplish a task quickly, so they will keep immediate production high by continuing to use a few basic problem solving procedures rather than mastering new, more powerful command sequences. Also, people often attempt to assimilate new phenomena by treating them as minor variations of already understood situations. For example, electronic desktop metaphors sometimes confuse users who find that computational "folders" differ in crucial aspects from their real-world counterparts.

In applications where direct manipulation strategies are not effective, a second type of self-explicating interface can aid in overcoming these production and assimilation problems. Second-person interfaces (in which a computational actor is told by the user to accomplish a task) can utilize intelligent explanatory abilities to handle situations where error, malfunction, or novelty is causing confusion. (These advice-giving interfaces are a subclass of the larger field of intelligent tutoring systems [Dede, 1986].)

A good review of interfaces which are effective in facilitating user learning is given in Burton [1986]. Historically, these explanatory environments have been difficult to construct, expensive, and limited by their specificity to a single domain. Researchers are now exploring intelligent authoring systems to solve this problem. Two examples are the Xerox Instructional Design Environment (IDE), which incorporates a hypertext representation structure [Russell, Moran, & Jordan; 1987], and the Teacher's Apprentice [Lewis, Milson, & Anderson; 1987]. Also, a space-related intelligent training system has been developed by the Artificial Intelligence Section at NASA-JSC [Loftin et al, 1988].

Advances in advice-giving interfaces depend on resolving several types of research issues. A general theory of coaching needs to be developed; everyone has experienced the irritating fact that poor or obtrusive advice is worse than none at all. Behavioral evaluation techniques for understanding the effectiveness of explicating interfaces are also currently inadequate. In addition, converting an expert system optimized for decision making to an explanatory function can be very challenging [Clancey, 1987]. Still, enough progress has been made on these challenges that this type of self-explication may be one of the next major applications for expert systems [Carroll & McKendree, 1987].

However, improving user learning through enhancing the system interface need not wait on advances in artificial intelligence. Simply presenting to users the sequence of actions taken in performing a task may suggest to them patterns of error or suboptimal performance. Such "cognitive audit trails" provide a means of documenting skill acquisition and support learning while doing. "Consciousness sensors" which monitor the user's motivation and mood through tracking respiration, skin conductivity, heart rate, and other physiological measures are another possibility for improving productivity through performance feedback [Dede, 1987b].

Resolving Ambiguity

Linearizing aids and second-person explanatory environments are interface design approaches which can reduce user confusion from the functional complexity and multiple simultaneous processes of knowledge bases. A final problem is that of ambiguity; interactive systems intrinsically depend on conversational postulates about the task being performed. The task mapping involved in moving from user goals (e.g. create a striking overhead transparency) to the actions the interface makes available can be challenging, and "sorcerer's apprentice" outcomes can easily arise from human actions misinterpreted by an information system.

One solution to task mapping problems would be natural language interfaces: a person's goal descriptions in a familiar communication medium are translated to command sequences by the machine. Such an approach is attractive because the interaction is built around already developed human skills. Also, knowledge about how to conduct clear conversations between people could be applied to user-computer interactions, and the expressiveness of natural language provides the foundation for a rich menu of possible commands.

However, substantial barriers exist to implementing true natural language interfaces. Natural language understanding has proven a very difficult problem in artificial intelligence, and major research issues are still unresolved [Perrault & Grosz, 1986]. Building contextual knowledge into an interface involves multiple, simultaneous levels of discourse [Reichman, 1987]. The implementation resources required for a natural language interface can be a major proportion of total system cost, yet the final product may still be quite limited in its capabilities compared to human conversation [Miller, 1986].

An alternative method for reducing ambiguity in task mapping is to make the interface an intelligent, semi-autonomous agent. One example of this approach is The Information Lens, an intelligent information sharing system [Malone et al, 1987]. Studies of knowledge transfer in organizations indicate that people use cognitive, social, and economic strategies in filtering incoming data. An intelligent information sharing system can use semi-structured (frame-based) message types which are automatically processed through intelligent editing mechanisms prespecified by users to accomplish this filtering. The use of an intelligent agent provides a substantial degree of user control in the human-computer interaction, as well as supporting group problem solving through an improved organizational interface for data management. In general, semi-structured interfaces seem a promising approach for task mapping.

Embedded intelligent agents could also improve human information retrieval capabilities. As Brown and Moskovitz [1985] discuss, secretaries are adept at finding documents in files even when the request includes erroneous information (i.e. wrong title or year). People use sophisticated, but largely intuitive search heuristics; if these could be incorporated into an expert system in a knowledge base, ambiguous or incomplete instructions could still be executed. In addition, a profile of the user's style in assimilating information could be included, so that requested data is presented in a format (e.g. tables, pie charts) comfortable for that person.

Even though natural language capabilities and semi-autonomous agents require substantial resources compared to "dumb" interfaces, the payoffs for building intelligence into an information system go beyond resolving problems of ambiguity. The intrinsic motivation of users to interact with a documentation system may depend in part on its ability to support perceptions of contact with an intelligent agent. At least seven types of motivations appear to sustain human interest in computer-based learning environments: challenge, fantasy, curiosity, and control are "individual" motivating factors; and cooperation, competition, and recognition are "interpersonal" factors [Malone & Lepper, 1985].

To the extent that a documentation interface can provide cooperation, competition, and recognition (either through intelligence embedded in the system itself or through computer-supported cooperative work features), user motivation to learn from its knowledge base will be enhanced. In general, designing interfaces to maximize motivation (e.g. mimesis as a means to enhance fantasy) may be as crucial as tailoring the user-computer interaction to minimize cognitive load.

Summary

The concept underlying a "cognition enhancer" is that of using the complementary cognitive strengths of a person and an information technology in partnership [Brown, 1985]. For example, computers have large short-term memories (megabytes of RAM), while human beings are limited to an immediate storage capacity of less than ten chunks of information [Anderson, 1983]. Computers can also execute complex "algorithms" (precise recipes for solving one class of problem) more rapidly than people. For tasks involving manipulation of successive symbolic results (e.g. involved mathematical calculations), these two cognitive attributes gives a computer an advantage over a human being. In general, computers are becoming superior at all forms of standardized problem solving.

However, as discussed in the **Alternative Knowledge Representation Approaches** section, people store information long-term in rich semantic networks containing webs of associationally related textual, temporal, and visual imagery. At present, computers are much more limited in how their information can be interrelated. Also, the cognitive attributes of human beings give them an advantage over computers at applying peripheral real world knowledge to ill structured problems, at problem recognition, at metacognition (thinking about thinking), and at non-standardized problem solving.

Researchers are working to develop stand-alone machine intelligence but, because of current limitations of computers, this task is very difficult. Cognition enhancers designed to combine the strengths of humans and computers will evolve much more rapidly. Advanced user interface approaches for semantic rationalization, coordinating multiple simultaneous processes, and resolving ambiguity build on this underlying concept of partnership between person and intelligent tool. As a result, new styles of learning and thinking may eventually develop from use of these "empowering environments" [Dede, 1987b].

Even stand-alone machine intelligence needs to be designed so that a rich interaction with human users is possible. Georgeff [1987] is conducting research on Procedural Reasoning Systems (PRS) which can be used to automate certain tasks characteristic of space missions. A major challenge in this work is creating computational modes of reasoning and planning that are efficient, extensible, robust, and dynamic--yet can serve as the basis of communication with humans: describing intentions, explaining reasoning, accepting advice. A sophisticated user interface is central to giving PRS systems these cooperative capabilities.

How are these general heuristics being applied to software production systems such as the SSE? Discussing the wide variety of new approaches emerging is beyond the scope of this review; a few which build on work discussed elsewhere in this study can be mentioned here. diSessa [1986b] has studied the mental models that programmers use. People find helpful both structural models (which deal with aspects of an object or action independent of specific use) and functional models (which deal with specific use, consequences, or intent). The stereotypes of the "formalist" and the "hacker" illustrate extremes of working with only with the former or the latter, and each type of model involves quite different interface characteristics.

In addition, people often engage in more informal cognitive activities (such as visual metaphors) that aid in coding. Any documentation environment for programming should have an interface flexible enough to support all these models. Boxer (discussed in the **Alternative Knowledge Representation Approaches** section) exemplifies this flexibility by using geometric and spatial relations; a mimetic, direct manipulation interface; device programming (creating functionality by assembling simulated components) ; microworlds; and capabilities for individual customization [diSessa, 1985].

Beyond mental model studies, researchers are analyzing users' conceptual knowledge of computer programming languages [Mayer, 1987]. Building on this work, several different approaches are being developed to provide programming system interfaces with embedded expertise (intelligent coaches which provide hints to optimize user performance). The Lisp Tutor [Anderson, Farrell, & Yost; 1984] illustrates a syntax-driven strategy of immediate error correction, based on a production rule model of cognition. In contrast, Proust [Johnson, 1987] infers the programmer's goals and methods from code semantics and provides a retrospective analysis of bugs and strategic errors. A variety of hybrid systems which combine the strengths of these approaches are likely to emerge.

Summary

Nickerson [1987] describes why the design of person-machine interfaces will be vital to space station productivity. Users need a documentation system interface which facilitates understanding and control so that they can recognize and correct unanticipated difficulties. Knowledge bases have characteristics of opacity, lack of causal metaphors, multiple simultaneous processes, functional complexity, and ambiguity which make designing such interfaces challenging. Promising directions include semantic rationalization strategies (mimesis, direct manipulation, microworlds), linearizing aids, explanatory second-person environments, natural language communication, internal semi-autonomous agents, and maximizing intrinsic motivation. Many of these strategies for cognition enhancement are being applied in exploratory research on advanced programming environments. This section has focused on interface design and the individual user; the next section expands this examination to how documentation systems can empower the task performance of groups.

Computer-Supported Cooperative Work

Computer-supported cooperative work (CSCW) is an emerging field which focuses on improving group productivity and effectiveness in shared computational environments. Research topics currently being investigated include collaborative design technology (surfacing collective assumptions, resolving conflicts, and developing shared models of a task) and organizational interfaces (text sharing, project management, and collaborative authoring systems). These build on the concept of multiuser interfaces (the integration of inputs from several users manipulating a common workspace). CSCW is a relatively new area of research--its first conference was held in December, 1986--but several themes of interest to SSP information systems have already emerged and are briefly summarized below.

Collaborative Design Technology

The Software Technology Program at the Microelectronics and Computer Technology Consortium has been exploring how Issue Based Information Systems (IBIS) can help software designers by supporting structured, collective conversations about planning [Conklin, 1986b]. A number of design activities can be improved through using such systems. First, collaborative information systems can aid with the coordination difficulties intrinsic in "wicked" problems (no definitive formulation, no stopping rule, "satisficing" solutions) [Rittel & Webber, 1973]. Space station applications are classic examples of wicked problems, so design tools for managing their cognitive and organizational complexity are very useful.

Second, such collaborative design technologies can capture the reasoning processes leading to a design decision. Recording the evolution of a design is important in facilitating documentation and in minimizing "backtracking" problems when a particular avenue of attack proves to be fruitless. All too often, the institutional memory of a rationale has eroded, forcing the project team to reiterate the design process. A prototypical documentation system for rationale capture is Xerox's Instructional Design Environment (IDE), which uses a hypertext representation system to build an explicit chain of reasoning between general learning theory principles and specific choices made by the designer in scripting a lesson [Russell, Moran, & Jordan; 1987].

Finally, computational support which models the design process itself can improve productivity and effectiveness. MCC has developed ISAAC, a hypertext format for structuring design decisions based on the following model:

1) delineate an issue;
2) develop a set of alternative resolutions;

3) analyze these competing alternatives on criteria such as cost, performance, reliability, and security; and
4) make a commitment to one alternative, with a subjective confidence rating on the correctness and stability of the commitment.

Sharing such models through linking individual hypertext representations can empower a design conversation in which competing approaches are explicitly presented and critiqued, with a collective mental model eventually being internalized by the entire project team. ISAAC is one of a set of integrated tools being developed as part of Leonardo, an overall MCC software design environment [Conklin & Richter, 1985].

Structured, machine-mediated communication underlies many types of CSCW. For example, as a step beyond electronic mail systems, computer-based teleconferencing facilities are being developed. These allow the real-time exchange of multimedia information, as well as access to a computational environment integrated across geographically dispersed information systems. Lantz [1986] presents a prototypical hardware architecture for structuring these networks through distributed computational support.

Once such a shared workspace is established, new types of task performance processes can be initiated. As an illustration, Lowe [1985] describes methods for cooperatively indexing, evaluating, and synthesizing information through multiple user interactions with a common database. In the Synview system, a structured representation is used to formalize group reasoning and debate processes about the accuracy and utility of alternative information sources. The argument structures themselves are stored as another dimension of knowledge encoded in the Synview system. This allows users to examine informed opinions about the quality of information they are obtaining. Such a CSCW approach can improve the accuracy, accessibility, and integration of knowledge bases.

Overall, while the technical and representational issues involved in implementing machine-mediated communication are significant, these collaborative environments offer several potential benefits:

- geographically distant people can exchange information readily,
- the documentation process (often the most neglected part of meetings) can be partially automated, and
- group interaction can be structured to be more productive.

Electronic mail and computer conferencing are preliminary steps toward achieving a "global village" within an organization; now, CSCW systems capable of realizing the other two benefits are emerging.

As an illustration, Xerox's Colab is an experimental meeting room set up to study how shared computational networks can be used to enhance problem solving in face-to-face group interactions [Stefik et al, 1987]. Colab generalizes the text editing concept of What You See Is What You Get (WYSIWYG) to WYSIWIS (What You See Is What I See); consistent images of shared information are presented to all participants in the meeting. WYSIWIS is used to support the decomposition of tasks into parallel activities which different members of the group can accomplish simultaneously in a common workspace.

For example, Cognoter is a Colab tool designed for collective preparation of presentations; its use results in an annotated outline of ideas and associated text. Meetings are structured into three phases--brainstorming, organizing, and evaluation--with Cognoter providing a shared workspace for recording and commenting on ideas. A direct manipulation interface is provided for entering, spatially rearranging, linking, and deleting information. An underlying hypertext representation allows the creation of semantic nets through transitivity and grouping operations. The outcome of a Cognoter session is a collective, fully documented mental model of a presentation.

Argnoter is a Colab tool used for presenting and evaluating design proposals; its focus is to facilitate discovering, understanding, and evaluating disagreements in a meeting. Personal attachment to certain positions, unstated assumptions, and unstated criteria often are hidden factors which cloud processes of cooperation, compromise, and group decision making. Argnoter segments meetings into three phases: proposing, arguing, and evaluating; the outcome is either a collective, documented resolution of the competing proposals or an explicit agreement to disagree.

Improving tools such as these will require research on both technical issues (such as the best control mechanism to use for mediating the interaction of multiple users in the same workspace) and cognitive issues (e.g. how the writing process can be structured to optimize group process during different phases of the meeting). This work in CSCW is crucial to improving the productivity of personnel who spend most of their time in meetings--such as managers--since standard office automation strategies have little impact on their task performance.

Organizational Interfaces

Malone [1986] draws a distinction between a user interface (which connects an individual to the capabilities provided by a computer) and an organizational interface (which connects human users to each other and to capabilities provided by the computer). Emerging examples of applications built around organizational interfaces include text sharing systems (electronic mail, computer conferencing), project management systems (which provide support for assigning people to tasks, constructing schedules, and allocating scarce resources), and collaborative writing systems (such as hypermedia authoring tools). In contrast to the **Organizational Impacts of Information Intensive Environments** section, which summarizes work on implementation strategies and the outcomes of technological innovation, research on organizational interfaces focuses on how to design CSCW systems so that they are easy to adopt and beneficial to use.

The design of organizational interfaces involves supporting types of problem solving behavior unique to groups [Corkill & Lesser, 1983]. For example, filtering inter-agent communication is not an issue in user interfaces designed for individuals, but requires careful computational control in CSCW systems to avoid an explosion of messages being passed among workstations. One approach to resolving this problem in an electronic mail system is the use of templates to structure different types of messages, thereby allowing the construction of intelligent filtering systems [Malone et al, 1986].

Also, goal conflicts among team members can adversely affect group problem solving; organizational interfaces can be designed to improve this situation through providing mechanisms for coalition formation and for confidentiality. Assigning subtasks to individual agents is another group function which CSCW systems must address; here, forming computer-based internal "markets" to bid on task assignments can be useful in enhancing institutional flexibility [Malone & Smith, 1986]. As discussed in the section on **Advanced User Interface Capabilities**, enhancing interpersonal motivations (such as cooperation, competition, and recognition) are also an important aspect of organizational interface design.

Filtering inter-agent communication, resolving goal conflicts, assigning subtasks, and intensifying interpersonal motivations are emerging interface functions intrinsic to group problem solving which researchers are just beginning to address. From this work, new strategies for improving collaboration in scientific research may result. For example, based on semi-structured interviews, Kraut, Galegher, and Egido [1986] describe a model of collaborative research relationships which involves stages of initiation, execution, and public presentation; evolution from one stage to another involves changes at both the human relationship level and the task level.

Analysis of the interview data reveals that establishing and maintaining a personal relationship is central to collaborative efforts in scientific research. Given the geographic and institutional barriers typical of very large scale development projects, a challenge for CSCW design is to create organizational interfaces which facilitate and support human emotional contact as well as the exchange of information. This is a very difficult task but, unless progress is made, potentially productive groups separated by accidents of distance or organizational structure may not coalesce into collaborative activities even though placed in electronic contact.

Summary

CSCW research builds on prior work in computer-based communications, computer-based information services, and computer-based decision support. One product of CSCW efforts is likely to be group decision support systems (GDSSs), which can be classified into six general categories [Kraemer & King, 1986]:

1) electronic boardrooms (display technology),
2) teleconferencing facilities (communications technology),
3) local area group nets (interactive conferencing capabilities),
4) information centers (knowledge bases and related tools),
5) decision conferences (structured decision models and protocols), and
6) collaboration laboratories (tools for joint authorship).

Research on these systems is evolving in two directions: studies of the nature of decision making, and developing technologically supported information systems to make group interactions more productive.

Long term, GDSSs may have different types of benefits. On the affective level, they appear to encourage the involvement of individuals in group processes and to build group cohesion. Organizationally, they can facilitate executing the protocols of group decision making, speeding the process of meetings. On the cognitive level, GDSSs can increase the quality of information teams use in reaching decisions or generating products.

Overall, work in CSCW is beginning to blur the distinction between two types of applications systems characteristic of information processing tasks [Danzinger & Kraemer, 1986]. Model-based systems flow from a decision support, operations research, management science framework; the design perspective underlying these information systems is that problem solving via model building is the core of task performance. In contrast, operations-based systems build upon concepts from computer science, data processing, and management information systems; here, problem finding is seen as the key activity which information systems facilitate. Through fields such as CSCW, synthesis applications are emerging which are data-based. These information systems are designed to facilitate browsing through data in search of facts, linkages, and patterns; both problem finding and problem solving functionalities are supported. TMIS, SSE, and other SSP information systems will need a collective, data-based perspective to support the full range of capabilities needed.

However, substantial barriers remain to the evolution of CSCW systems. Technical problems include requirements for accessibility and flexibility in computing resources, display and graphics technology limitations, and the shortcomings of current software for modeling and analysis. An incomplete understanding of the decision making progress is a cognitive barrier to the utility of GDSSs. Finally, human resistance to change in familiar operating procedures is a formidable obstacle, as discussed later in the section on **Organizational Impacts of Information Intensive Environments**. First, however, the potential availability of hardware capable of supplying the power needed for these advanced functionalities (knowledge representation, user interfaces, CSCW) must be examined.

The Evolution of Information System Hardware

Information technology has historically experienced exponential rates of change, and this situation seems likely to continue for at least another decade before fundamental physical limits (the speed of light, quantum mechanical effects, entropy) slow the rate of advance of electronics. Whether rapid evolution continues beyond that point will likely be determined by progress in optics, biotechnology, and computer architectures that go beyond the Single Instruction stream, Single Data stream (SISD), von Neumann model. Mid-range trends for computer hardware can be summarized as higher capacities; increased abilities to represent complexity and encompass a wide range of applications; and faster, smaller, and cheaper systems.

This section first analyzes the types of system development problems that can arise from rapid hardware evolution. Next, an overview of management strategies for addressing these problems is presented. One key approach is strategic planning; to enable this activity, forecasts of advances in computers, secondary storage technologies, telecommunications, and systems integration are delineated. Finally, the section concludes with a timetable for likely advances in information technology and a discussion of its implications for NASA schedules.

In general, technological evolution is considered a positive factor from a functional capability perspective. However, it can often result in costly side effects for large scale, capital-intensive projects such as TMIS and SSE. These difficulties manifest themselves in such forms as:

- premature obsolescence of existing systems as their underlying hardware base is no longer supported
- functional incompatibility of components between generations
- repeated training of human resources on successive generations of technology
- sophisticated systems which can be more susceptible to the negative effects of unexpected transients
- difficulties in anticipating the final design of a system being built over a long time period

Problems of Rapid Hardware Advances

Premature obsolescence results from the introduction of new technology which surpasses the capability of its predecessor before amortization of the initial investment has occurred. Also, successive generations of hardware may be difficult to link with earlier systems, particularly where a discontinuous change has occurred in the design paradigm which underlies the technology. For example, software written for the first generation of eight-bit, CP/M based personal computers is not compatible with the standard MS-DOS, sixteen-bit PC of today.

Technological evolution must also be accompanied by ongoing training of human resources in order to make the best use of new capabilities. As an illustration, the level of user understanding needed three years ago to draft and print a hardcopy of a document using a word processor was significantly less than now necessary for the task of creating a compound document on a desktop publishing system.

In addition, sophisticated systems are more difficult to fix when a problem does occur. This does not imply less reliability of the system, but technological evolution tends to produce more complex devices requiring increasingly specialized problem solving resources to repair. For example, compare the skills and tools necessary for maintaining today's automobile versus a car of fifteen years ago; pliers and screwdrivers have been augmented by diagnostic computers. Finally, anticipating the terminal design of a system which is evolving over a long period of time is often considered a "wicked" situation, where the outlines of the solution can only be deduced by actually solving the problem.

Other types of hardware concerns stem from pressures for cost effectiveness and for system evolution which builds upon NASA's historic base of computational architecture. TMIS and SSE present one of the greatest technological challenges conceived by modern man; the level of sophistication needed to achieve Space Station Program objectives is such that, in many cases, requirements exceed today's leading edge research prototypes. However, SSP systems are to be built using Commercial-Off-the-Shelf-Technology (COTS) wherever possible and must link with systems which date from space projects undertaken several decades ago.

Acquisition of COTS, while contributing to SSP directives of cost effectiveness and immediate initiation of activity, could undercut NASA goals of scientific leadership and the creation of useful spin-off technologies. COTS is designed by a wide variety of companies motivated by incentives beyond user productivity (e.g. planned obsolescence, competitive uniqueness); this typically results in unpredictable, often mutually exclusive evolutionary paths. Important underdeveloped elements of NASA management strategy are well established protocols for determining trade-offs between cost and capability and for deciding among priorities when inter-goal conflicts occur.

Further, NASA funding is controlled through a politically oriented yearly budget cycle in which optimizing capital expenditures is essential and short term accomplishments in the complex arena of space development weigh heavily upon future funding decisions. To develop a system such as TMIS or SSE over an extended time frame, using cost effective approaches and maintaining compatibility between phases, requires an evolutionary path which builds upon existing system components at each stage of development.

The result of addressing these complex requirements and objectives in the face of rapid technological advances will likely be a system which is created in an incremental, yet disjointed fashion. Disjointed incremental design is not an appropriate paradigm for engineering a complex system of the magnitude of Space Station; however, no alternative appears feasible at this time.

Managing Hardware Evolution

A variety of methods are used to combat problems of large scale systemic complexity and uncertainty caused by extended time horizons and rapid rates of change. At the conceptual level, methodologies such as forecasting and technology assessment (which are operational requirements for the prime TMIS contractor) may be used to identify the costs and benefits of alternative plausible paths for system development. Also, systems to optimize project scheduling, control, and configuration for large engineering endeavors are being developed [Sathi, Morton, & Roth; 1986].

At a strategic level, processes in place to aid in managing information technologies include the development of computing and telecommunications standards. These activities provide general guidelines for the design of information and telecommunication technologies which support connectivity and compatibility guidelines. A perspective on these efforts is presented in the section following this, **Standards, Connectivity, and Compatibility**.

At a tactical level, hardware and software vendors generally provide for a limited degree of upgradability and compatibility within their system families through maintenance agreements and integration products. These agreements and products are designed to provide problem solving assistance and enhancements for that vendor's hardware or software technology over a limited time frame.

Illustrative of this type of effort are certain IBM products. Between the mid-1970's and the mid-1980's, IBM offered many types of Central Processing Units (CPUs) designed to respond to the varying computing needs of large corporate environments. These CPUs included the 43XX and 30xx, 370-architecture mainframe computers; the 8100, a distributed minicomputer; the Displaywriter™, a dedicated word processor; Systems 36 and 38 (3X), multiuser minicomputers; the Scanmaster™, a digital scanning system; the Personal Computer™ series (8088, 8086, 80286, 80386); and the 5520, a multiuser office automation system which may best be categorized as a supermicrocomputer.

Many large organizations implemented some portion of these systems within their environment. However, due to incrementally disjointed design, each of these CPU model lines had its own unique hardware and software architecture. The result was an inability to link these various systems in order to communicate data files in a common format and share functional capabilities.

The solution posed by IBM was DISOSS (a distributed office system software solution) which, when used with Systems Network Architecture (SNA™), was to interconnect all these devices and provide revisable format document connectivity. The success of the DISOSS approach to date is questionable, as retrofitting compatibility is a difficult task. However, providing a software-driven umbrella architecture which makes heterogeneous hardware and software components in a large information system environment homogeneous is a mandatory requirement of future large scale information systems.

Also, Digital Equipment Corporation (DEC) has long touted compatibility and upgradability as a major capability of its computing systems. Interconnectivity among Digital computing hardware is embodied in the use of a common operating system (VMS™) throughout their VAX product line. Application software and peripherals generally operate on any VAX CPU, thus providing portability. VAX CPUs may also be clustered together to build large, distributed data processing networks.

In summary, the hardware substrates necessary to fulfill TMIS and SSE requirements will not be single vendor solutions, but will require systems integration. Present design criteria for both TMIS and SSE require multi-purpose hardware and software. For example, an SSE terminal may be used for creating documentation requiring the manipulation of mixed object documents (text, graphics, and images) or for software development or for access to relevant databases and telecommunication networks. From a management perspective, successful synthesis over an extended time frame necessitates applying not only methods and processes currently available to assure multiple-vendor connectivity, compatibility, and upgradability over time; but also the development of new paradigms for large scale systems engineering.

Strategic Planning for Hardware Developments

Another management issue stems from questions about the timing of advances in information technology. The **Introduction** section presented a vision of what capabilities a future SSE workstation might possess. Three key sectors which must progress to provide the power necessary for these functionalities are computer architectures, secondary mass storage systems, and telecommunications. Forecasts of likely developments in these areas over the next two decades are given below.

The ultimate accuracy of any statement about the future ultimately depends on making correct initial assumptions about the dynamics of change. In particular, the evolution of information technology has a number of driving forces which affect both speed of change and level of discovery. An influential enabling force is the level of spending institutions allocate for research and development.

Federal R&D spending for basic and applied research in information technology related fields (including computer science and electrical engineering) increased from approximately $371 million in 1976 to over $933 million in 1986 [Office of Technology Assessment, 1985a]. 1986 R & D spending for all categories of research by private sector U.S companies totaled $51 billion (of their own revenues). Combined with federal spending (for all categories of research), $65 billion dollars (or approximately 2.5% of the Gross National Product) were expended. This represented an increase from 1985 R&D allocations of 16% for the computer industry and 17% for the software industry. The forecasts below are based on the assumption of general, gradual increases in these research and development expenditures.

Advances in Computer Architectures

Today's computing systems are still based primarily upon the von Neumann model, in which a single, central processor executes instructions on data in a sequential or linear fashion. Advances in the von Neumann architecture have given a desktop PC the equivalent power of a large 1960's era mainframe system at a miniscule fraction of the cost. Illustrative of recent changes is a comparison of four generations of Digital Equipment Corporations MicroVAX™ computers.

MICROVAX SYSTEM COMPARISON

	Date	MIPS	Memory	Price ($)
MicroVAX I	10/83	0.36	2-5 MB	12,800
MicroVAX II	5/85	0.90	2-9 MB	14,000
MicroVAX 2000	2/87	0.90	4-6 MB	7,500
MicroVAX 3000	9/87	3.00	16-32 MB	75,000*

* Includes a license for 20 Users

Increasing functional capability and decreasing physical size and cost are the result of technological evolution. Such benefits are achieved by changes in the microelectronics which comprise the system; however, engineering is approaching the limits of computational speed achievable with current silicon design.

One billion components per chip is the estimated state-of-the-art for the year 2000 [Meindl, 1987] and, as gigascale integration is reached, the rate of increase in component density will slow. For example, components per chip increased by three orders of magnitude in the 1960's, and by two orders of magnitude in the 1970's; less than an hundred-fold increase will occur in the 1980's, and the 1990's may see a gain of only ten-fold. Future significant leaps in computer performance will require advances in parallel processing architectures, large scale integrated circuit design, new chip substrates, and exotic technologies such as three-dimensional quantum-well superlattices.

Integrated circuits and microprocessors are the heart of any computing system. At this level, two major determinants affecting functional capabilities are signal processing speed and switching speed. Signal processing speed, or the rate at which data can be processed (how fast the chip can put through bits of data), is linked to the number of components on a chip, chip size, and the substrate material used in construction. Switching speed (how fast data can be sent through a circuit from one component to another on the chip or board) is linked to board design, chip size and packing density.

Signal processing speed has been increasing as a result of greater circuit density and improvements in chip substrate and construction technology. Presently a one square centimeter chip contains one million (1 Megabit) components. Near-term extrapolations indicate 4 Megabit chips will be commercially available by 1990, and somewhere between 10 and 100 million components per chip will be state-of-the-art by the year 2000.

Density approaching the 100 million component per chip range would require line widths of 0.1 microns (in contrast, the Intel 80386™ is a 1.5 micron chip) and chip sizes of about 1,000 square millimeters. In order to accomplish such breakthroughs, microlithography (the underlying manufacturing process of "implanting" chips) would have to employ the use of electron-beam and ion systems.

Evolution in the basic materials used to construct chips will further contribute to the trends of increasing speed and decreasing cost. New materials {e.g. gallium arsenide (GaAs) and gallium aluminum arsenide (GaAlAs)}, coupled with new techniques in crystal construction such as heterostructures and ballistic transistors [Brody, 1986], will result in improvement in the crystal's carrier mobility (electron speed through the circuit layers). Electrons in a heterostructure have been clocked at up to 20 million centimeters per second (cm/s), versus 6-8 million cm/s for an ordinary silicon device. To give a sense of what this means for processor speed, the limit for silicon transistors is around 20 GHz; and conventional GaAs, about 50 GHz. General Electric has demonstrated a heterostructure transistor which operates at frequencies of up to 80 GHz.

Improvements in chip fabrication methods, wafer size, circuit density, and substrate compounds will all contribute to dramatic improvements in chip size and speed through the end of this century. Realizing the full benefits of gains in the signal processing capabilities of chips require similar improvements in methods of interconnection or switching technology between processors in order to result in faster computer architectures. Recent advances in superconductivity and optical signal switching hold great promise for increasing these inter-chip capabilities.

Significant breakthroughs in superconductive material development have been achieved within the last eighteen months. Once perfected, these new materials, which can transmit electrical energy with no resistance, could be applied to such areas as "interconnects" (the circuitry, now typically composed of aluminum or copper, which links computer chips together). The potential results are a decrease in power dissipation and an increase in signal transfer rate.

On the horizon are functional prototypes for an optical computer (optoelectronic) which utilizes laser light to pulse data between chips at rates of up to one gigabyte per second. The optoelectronic device uses coherent laser light to replace physical circuits and gallium arsenide chips in order to produce the superfast integrated circuits required for parallel processing applications.

Three major benefits of incorporating optoelectronic devices are [Hutcheson, 1986]:

- increased density (greater computing capacity in a smaller physical area)
- lower parasitics (a reduction of the negative effects of electrical resistance and heat generation within the processor), which means higher performance
- reductions in the number of parts, easing packaging

Switching speed has been improving (from 1 millionth of a second in mid-1960's, 1 billionth of a second in 1977, to an expected 10 trillionth of a second by 1990) and is expected to approach 1 trillionth of a second by the end of this century [Office of Technology Assessment, 1985a]. However, this trend of increasing device speeds for both silicon and gallium arsenide components will slow towards the latter part of the 1990's as the limits of these media are approached.

In summary, microprocessor technology is a fundamental building block of creating advanced computing systems. While conventional architectures will continue to improve, parallel processing (in which multiple, linked processors execute many software routines simultaneously) may be necessary in order to provide the power needed for advanced user interfaces, expert systems, three-dimensional displays, the manipulation of massive volumes of digitized information and emerging forms of knowledge representation. Work on such connectionist architectures is proceeding based on electronic simulation of neurobiological models [Hopfield & Tank, 1986], massively parallel processors arranged in a Boolean *n*-cube [Hillis, 1987], optical neural computers [Dongarra, 1987], and many other variations. The equivalent of parallel processing in software design is also emerging [Hillis & Steele, 1986] and may be required for the types of functional performance SSP requires.

Advances in Secondary Mass Storage Devices

The information contained within TMIS is conservatively estimated to average 250 gigabytes of core knowledge annually. The design and development of SSE software for Space Station operation is expected to require tens of millions of lines of code. TMIS and SSE documentation requirements including accessibility, distributed processing, multi-media formats, and large volumes; this will necessitate the use of a variety of memory storage devices.

Computing systems make use of two kinds of memory storage. Random Access Storage (RAM) is chip-based, primary computer memory used to execute software and process data. In contrast, permanent, secondary storage encapsulates large volumes of data either in a magnetic medium, such as disks or tape, or in a optical medium (e.g. a compact disc).

Secondary computer storage systems have historically utilized magnetic disks or tapes. Storage capabilities range from approximately 0.5 megabytes (or roughly 100 pages) for a conventional 5 1/4 inch floppy disc to 1000 megabytes for the large hard disk packs used with mainframe computers. Magnetic storage relies upon a thin film media and a "head" reader which travels over the surface of the disk (which, in the case of rigid hard disk systems, rotates at speeds of 3600 rpm). The limits of conventional magnetic media are expected to be reached within the next decade because of physical constraints in density packing on thin film and the attainment of minimum head reader heights over thin film media [Bezold & Olson, 1986].

Optical storage technologies present the latest wave of mass storage breakthroughs: Compact Disc-Read Only Memory (CD-ROM), Compact Disc-Write Once Read Many (CD-WORM), and Compact Disk-Interactive (CD-I). CD technology uses lasers to store data in a digitized format on an optical disk medium ranging in size from 4.7 to 12 inches in diameter. Storage densities for CD-ROM, which is used to archive permanent information, range from 500 megabytes to 1 gigabyte per disc. The Write-Once-Read-Many-Times (WORM) format allows the user to store textual, numeric or image information a single time. A CD-I system expands this capability into storing alphanumerics, images, graphics, or audio/visual formats all on the same disc; however, with the CD-I format, the "write" technology is generally not available to the average user, as it requires specialized equipment.

Access time for retrieval of information from optical storage has improved from 400 to 500 milliseconds to 200 to 300 milliseconds, but this is still not as fast as magnetic storage systems, which can retrieve information in the 18 millisecond range. Optimization of compact disc access speed is still evolving, and software driven optimizers (such as new indexing techniques) are expected to contribute to improvements.

A number of commercial optical storage products have been recently introduced. For massive archival information, CD-WORM disks have been packaged in a "jukebox" system. The player holds a number of optical disks, which store multi-media information formats for on-line access. One such system is the Optical Disk Storage and Retrieval Unit™ (ODSR) made by Perceptics International (formerly Optical Storage International).

This system contains 20 double-sided, removable WORM disks, each of which stores up to 2 gigabytes of information. The system cost is about $65,000. The ODSR can be linked to an individual computer or a network through the Small Computer Systems Interface (SCSI) or the Intelligent Systems Interface (ISI). (The SCSI and ISI are industry standard interfaces which link optical storage devices to computing systems.) Up to seven ODSRs can be linked together, creating a potential of 320 gigabytes of storage.

One of the problems raised by implementing an optical mass storage and retrieval system is the task of capturing existing information in an electronic formation suitable for storage. Some optical systems store information as an image rather than in a revisable format. This may be appropriate for archived information which does not change, but is merely retrieved for historical purposes; however, dynamic information which is part of a production process must be stored in a revisable, updatable form. While it is possible to optically store information in a revisable format, the WORM medium limits updating existing information. This limitation will be resolved by the next generation of optical storage (termed "erasable" optical storage) expected to emerge by the early 1990's.

A second challenge created by the implementation of optical storage is the conversion and capture of information which exists in a non-magnetic format, such as paper or micro-fiche. For example, in SSP Phases 0 and 1 generated thousands of megabytes of conceptual design and requirements information which lay the foundation for much of the overall project. Despite a preliminary baseline configuration for TMIS which theoretically generated SSP materials in a standard electronic format, many of these documents are unavailable in anything but hardcopy. The conversion of these documents to a suitable electronic format for archival purposes may be challenging.

However, one promising strategy for resolving this problem is through the use of optical scanners, such as the Palantir Compound Document Processor™. These devices digitize text and images by scanning paper documents into a revisable electronic format, which can then be treated as a conventional electronic file for storage and retrieval.

Significant improvements which optical storage offers over conventional paper/microfiche mass storage systems include:

OPTICAL VS. CONVENTIONAL MASS STORAGE SYSTEMS

	Optical	Paper/Microfiche
Storage Density	200,000 pages/disc	3000 pages/drawer 260 pages/fiche
Storage Speed	Computer speed	By hand
Storage Cost	\$0.008/pg.	Micromedia \$0.01/pg.
Storage Space	200,000 pages on one 5" disc	200,000 pages in 180 file drawers
Retrieval Time	3 to 5 seconds (random access)	minutes (sequential access)
Indexing	Automatic Multi-level	Manual Limited levels
Shelf Life	15 year minimum possibly indefinite	100 yrs with proper processes, medium, and storage environment

Economies of scale in the production process will continue to drive down the cost of optical disc and disc reader technology.

Future innovations in storage technologies will likely stem from two advances:

- The development of an inexpensive "erasable" optical storage medium
- The use of new types of energy devices to "burn" data into the recording medium

Early versions of erasable optical memories have recently emerged in the marketplace. These initial entrants generally make use of magneto-optical or phase changing media in order to provide the capability to read and write over the same medium numerous times [Tsunoda, 1985].

These erasable systems represent the first generation of optical disk technology which will be able to compete with the multiple read/write capabilities of magnetic systems. At present, these optical products are not commercially competitive because of cost; limited production facilities for erasable systems do not enjoy the same economies of scale as the more mature magnetic media. However, commercial penetration of erasable optical storage is expected to emerge in the early 1990's and to expand substantially over the decade following. Second generation erasable optical storage technology will make greater use of phase changing materials, which provide higher speed and disk densities than first generation magneto-optical erasable technology.

Other expected advances in optical storage include the use of High Power Diode Lasers, which use a shorter wavelength to increase bit density on the storage medium. The shorter wavelength essentially allows smaller (and thus denser) bit hole patterns. Frequency Domain Optical Storage, another laser based system, permits the lasing light to be adjusted over a range of frequencies, thus allowing information bits to be overlapped at different frequencies. The data is then retrieved by adjusting the laser to the specific frequency originally used during the write process. This innovation could potentially increase the capacity of optical disks by a factor of 10 to 1,000 [Office of Technology Assessment, 1985a].

Electron Beam recording for optical media is now under development. Electron beams operate at a substantially higher frequencies and shorter wavelengths, thus providing even greater bit densities than lasing light technology. Error rates are still unacceptably high; however, given the capability to control errors through error detection algorithms, E-beam could boost storage by an order of magnitude into the 50 gigabytes per disc range [Bezold & Olson, 1986].

In summary, mass storage devices have begun a shift toward the use of optical technology, although magnetic media will continue to play a role because of rapid access times. As the implementation of TMIS and SSE proceeds, developing an integrated system of magnetic and optical storage devices will be an important priority in managing the high information volume of these systems.

Advances in Telecommunications Technology

Telecommunicating via electronic means over distance is an important requirement for the Space Station Program. NASA centers and laboratories, SSP contractors, and international partners all have the need to send and receive large volumes of information. Transmission of data will be required on a local level (such as on-site at the Johnson Space Center), throughout metropolitan systems, (i.e. between NASA Centers and surrounding contractors), nationally (routes between cities to link NASA Centers), and transoceanically (international partners).

This information takes a wide variety of forms: audio, video, images, graphics, text, and numbers. Volume data transmission with many concurrent users is a central requirement for both the Program Support Communication Network (PSCN) and TMIS, which together comprise the primary telecommunications network for the Space Station Project. SSP telecommunication requirements are expected to encompass the full range of communication technology, from local area networks to international teleconferencing.

Telecommunication of information is accomplished through either land-based or space-based systems. These are generally configured as:

- Digital microwaves via land-based transmitter/repeaters
- Land-based cables (coax-copper and fiber optical)
- FM radio transmissions
- High frequency Ku and C band satellite transponders

Telecommunications capacity, or the capability to send greater volumes of information at a faster rate over distance, has been increasing at unprecedented rates. The following exhibit illustrates the increases made through improved technology for telecommunications.

INCREASING TELECOMMUNICATIONS CAPACITY

300 bps (bits per second)	Considered high speed data transmission at start of 1980's
1.2-2.4 kbps (kilobits/sec)	Data transmission speed common now
1.54 mbps (megabits/sec)	Telephone network T1 carriers
6 mbps	Today's satellite transponders
16 mbps	Extremely clean 4-wire copper circuit under ideal conditions (not usually available)
42-160 mbps	Today's fiber optic technology
2 gbps (billion bits/sec)	Rate recently achieved by Bell Labs for an 80 mile fiber optic transmission using no amplification

Two major technological trends for telecommunications are expected to continue through the next decade:

1) increasing transmission capacity; and
2) standardization of heterogeneous hardware and software, resulting in improved integration among different vendors.

Increasing transmission capacity will result primarily from technological improvements in fiber optic cabling and satellite transponders. Fiber optic cabling is composed of ultrahigh purity glass fiber lightguides. Flexible, strong, and lightweight, these fibers carry audio, video, or data transmission in the form of coherent light pulses generated by diode lasers. Fiber optics are not adversely affected by electromagnetic fields, can carry orders of magnitude more information than a copper co-axial cable, and require fewer signal repeater between spans than copper cabling. For example, a quarter-inch diameter optical cable with two fibers carries as much data as a 3-inch copper cable with 20,000 wires.

Fiber optic cabling is presently being employed for transoceanic, long haul domestic trunks, and metropolitan systems. Illustrative of this is a recent Bell system long-haul system: a 900 mile, single mode lightguide which operates at 432 mbps. This network carries up to 6,048 voice circuits over a single fiber in each direction [Kay & Powell, 1984]. These systems are replacing amortized co-axial systems and, by the 1990's, are expected to begin penetrating into residential usage. By the end of this century, fiber optics will be the dominant transmission medium for all fixed applications.

Additional benefits of fiber optics include a relatively short signal delay time, dissemination of cabling along existing (co- axial) paths, and greater security than radio or satellite transmissions. The capacity of fiber optic technology has been doubling annually for the past decade, and these gains are projected to continue through the end of this century. Such improvements--which are the result of new plastic materials, reduced fiber impurities, and improved fiber splicing techniques coupled with breakthroughs in new lightwave technology--could bring fiber optic transmission technology to its physical limit of 10^9 mbps per kilometer [Office of Technology Assessment, 1985a]. By the year 2000, speeds of ten gigabits per second on a single fiber seem attainable [Kahn, 1987].

Networked computer architectures will need transmission rates of this order of magnitude to accomplish the types of functions necessary for SSP. A computer terminal with a color 1000 X 1000 display, transmitting at 30 frames per second, requires 480 megabits per second of network transmission speed. Bigger displays (i.e. four feet square) could increase transmission requirements by another two orders of magnitude in real time situations, even with data compression. The cost of these large, flat panel monitors is dropping steadily, although the number of edge connectors required makes quantum leaps in performance difficult, and which competing medium for the display (e.g. plasma, crystal) will dominate is still uncertain.

The other major factor in increasing transmission capacity, satellite telecommunications, experienced rapid growth in the 1970's, as business and military use flourished in the United States and the Soviet Union. Internationally, access to satellite technology is made available to nations through the International Telecommunications Satellite Organization (INTELSAT).

Satellite communications are accomplished by high frequency (C and Ku band) radio transmission to geosynchronous satellites operating in an orbit of approximately 22,300 miles above the earth. Usage of this communications technology has grown rapidly over the past 20 years, due to improvements in satellite channel capacity and reliability (through better antenna structures, power amplifiers, low-noise filters, and space-proven prime satellite power systems). Expected gains in bit rates and growth in capacity over the next ten years are given below:

SATELLITE SYSTEM BIT RATES

* 56 kbps for data systems
* 1.544 or 2.048 mbps for data channels
* 6.44 or 8.448 mbps for high speed data channels
* 16, 32 and 64 kbps for voice channels
* 1.544 or 2.048 mbps for one-way video conferencing
* 56 or 64 kbps for freeze-frame video

Source: [Kay & Powell, 1984]

GROWTH IN SATELLITE CAPACITY

(Equivalent 40-MHz Transponders)

Year	World	United States
1980	426	156
1985	1,410	478
1990	3,100	756
1995	5,580	1,135
2000	9,870	1,655

SOURCE: *IEEE Communications*, May, 1984

Continued improvements in multi-beam satellite antennas, on-board satellite switching, high frequency/high power systems, and lower cost earth stations will further contribute to growth in the satellite communications market. Illustrative of research to improve satellite communications is the NASA Advanced Communications Technologies Satellite (ACTS), which is intended to develop technology for a high frequency/high power Ku band (30/20 GHz) satellite system [Office of Technology Assessment, 1985b].

The use of satellites as a communication system is expected to continue through the end of the century. However, the rate of this growth is subject to a number of technological and economic variables arising through competition from fiber optics. Satellite communications has several inherent characteristics which make fiber optic cabling an appealing alternative. The effects of external influences such as electromagnetic disturbances and weather on satellite communication, cost factors, security issues, and technological complexity may cause land-based fiber optics to dominate communications systems in the future.

Satellites are dependent on complex underpinning technologies such as launch vehicles and space-based propulsion and power systems. The economics of fiber optics are expected to eventually exceed satellite systems, with the possible exception of very long haul communications, because transmission cost by satellite is nearly invariant with distance, while transmission cost by cable is not. Even the long haul economies of satellites over cable are ultimately questionable because, while transmission cost for cable increases with distance, it does not increase proportionally for long cables.

Thus, one major technological trend for telecommunications is increasing transmission capacities through fiber optic cabling and satellite transponders. A second major technological trend in telecommunications is the development and implementation of standards. Examples are 1) the OSI seven layer model (a hierarchical structure of communicating peer protocols which sets the future standards for computing systems hardware and software) and 2) the Integrated Services Digital Network (ISDN), envisioned to be a digital, global, public communication network which standardizes digital communication interfaces (facilitating future connectivity of communication networks regardless of geographic location and vendor). This very high bit rate telecommunications transport system encompasses personal message services, facsimile, teleprinting, computer-to-computer transmission, word processing, information retrieval, monitoring, transactions and teleconferencing [Kay & Powell, 1984]. Both these concepts are discussed in greater detail in this study in the **Standards, Compatibility, and Connectivity** section.

Initial standards for connectivity created through the development of the OSI Seven Layer Model and the ISDN have just begun to emerge in hardware and software technology. By the mid-1990's, a significant proportion of computing and telecommunications systems from major vendors will have implemented these voluntary connectivity standards. By the end of the 1990's, OSI standards are expected to be mature, and the majority of computing systems and telecommunication networks based on these will be in advanced stages of implementation.

In summary, telecommunications is expect to continue rapid growth from both a speed and capacity perspective. The dominant transmission technology through the end of the 1990's is fiber optics, which will be used for metropolitan, domestic long-haul, and transoceanic transmission. Fiber optics is replacing co-axial and twisted pair copper cable and is becoming economically more competitive than satellite systems, the next most prevalent communication system to be employed through the end of the century. Ground based microwave transmission and FM radio communication (cellular radio) will continue to be used for selected applications.

Emerging world standards for digital communication networks and computing systems will foster growth, as historically domestic economies move toward a global marketplace and become more reliant on voice and information processing technology. Current trends can be expected to create dramatically improved connectivity of information technology within multi-vendor environments and increases in the range of services and capacities of telecommunications mediums. The result will be decreased costs stemming from economies of production.

Hardware Integration and Long Range Evolution

One barrier to information technology utilization has been incompatibilities among different devices. The challenges currently involved in interconnecting computers, peripherals, and telecommunications systems can be staggering. However, this situation is slowly improving, for three reasons. First, emerging generations of computers are sufficiently powerful to emulate each other: for example, a software program now allows a Macintosh II to run IBM PC compatible applications. Second, as discussed above, standardization efforts are gradually creating uniform protocols for allowing devices to work together.

Third, and ultimately most important, computers and telecommunications will gradually merge into "synthesis technologies." The increasing universality of digital code and the dropping cost of powerful processors empower any device to have embedded microchips dedicated to interconnecting with other media. Within a generation, the distinction between computers and telecommunications is likely to be obsolete, and building a network of devices from different vendors will no longer be a major challenge.

Even with current technology, new types of interconnections are emerging. For example, a device is now available which allows a microcomputer to accept high resolution video input at thirty frames per second and store that data digitally in real time. This will facilitate the linkage of computers to inexpensive, charge coupled video camcorders and opens up a broad spectrum of possible mixed media applications.

Synthesis technologies are also evolving in peripheral devices. For example, laser printers are gradually dropping in price, expanding in capability, and seem likely to replace competing technologies (dot matrix, ink jet, thermal, impact) within the next decade. As a result, the "laser copier" may displace the standard xerographic equipment used now for hardcopy reproduction. Such a device would combine an optical scanner, RAM storage, and a laser printer to eliminate the mechanical collator. A series of pages could be scanned in and printed out already collated, since all copies of a particular page no longer need to be made at the same time!

These types of technologies are driven by falling prices. For example, the cost to vendors of optical scanning elements capable of three hundred dots per square inch resolution has dropped from two thousand dollars each in 1983 to seven dollars. Hand-held copying devices (which output a strip of tape that later can be inserted in a laser printer to produce the hardcopy) are beginning to emerge in response. Twelve hundred dot per square inch scanners are expected in the next few years; these will empower devices competitive with typesetting.

As a longer range issue, in about fifteen years the limits of electronics as a substrate for information technology are likely to be reached. The speed of light forces the size of faster computers ever smaller (for example, electricity travels about fifteen centimeters in a billionth of a second, so devices that switch at that speed cannot exceed that size). At dimensions this small and with the number of components per chip exponentially increasing, the heat generated by entropic effects from hundreds of millions of transistors rapidly switching in a tiny volume poses profound dissipation problems. Also, when extremely small distances separate different channels in a chip, quantum mechanical "tunneling" of electrons from one channel to another--regardless of the insulating properties of the intervening material--creates severe packaging challenges.

Therefore, while improvements in parallel processing and software engineering may continue to increase the effective functionality of electronic devices, further rapid advances in power and cost effectiveness for linear, sequential architectures will require a shift to an alternative medium (probably optical or biological in nature). Purely optical computers have theoretical limits about three orders of magnitude faster than electronic devices, and laboratory work with "transphaser" crystals capable of amplifying laser light emissions is underway [Abraham, Seaton, & Smith; 1983]. Major challenges involve developing, on a purely optical level, equivalents to electronic capabilities for random access memory, read only memory, multiplexing, etc. Advances in optoelectronics are driving much of the progress currently being made, but what barriers may later be encountered is currently uncertain.

As an alternative, emerging capabilities in genetic manipulation may allow the construction of "biochips," which would contain transistors as small as individual molecules [Drexler, 1986]. Such a development could be possible through using tailored enzymes and variations on monoclonal antibodies to deposit three-dimensional patterns of semiconducting materials [McAlear & Wehrung, 1984]. The resulting chip (nanotechnology) might be composed primarily of organic compounds and could conceivably use neural networks similar to those in the human brain as its primary processing architecture.

Whether or not such exotic information technologies are feasible, pushing electronics to its limits over the next several decades is likely to produce the equivalent of today's supercomputer power at desktop computer prices. Such an evolution opens up a vast array of new types of applications [Office of Technology Assessment, 1985]. For example, a problem that takes one and one-half years to solve on a current microcomputer could be completed in fifty hours. A calculation requiring four days would take thirty minutes; a six minute problem, less than two seconds. The power to implement emerging functionalities in knowledge representation, user interfaces, and computer-supported cooperative work will eventually be inexpensively available.

Timetable

To give more detailed information on the timing with which the diverse capabilities discussed in all sections of this report might emerge, a table is presented below which, for different functionalities, estimates commercial availability at prices comparable to advanced personal computer costs today. These projections have significant margins for error, since technical forecasting is ultimately speculative rather than factual, and even experts in a particular specialty often disagree about when a capability may be achieved.

Functionality	Uses	Time Frame
High resolution color monitors with 3-D graphics	Vivid simulation of reality; easy reading of text	Late 1990's
High bandwidth fiber optic networks	Massive real time data exchange	Early 1990's
Standardization of computer and telecommunications protocols	Easy connectivity, compatibility; lower costs	Mid 1990's
Optical disk systems with multiple read/ write and mixed media capabilities	Support of large data and knowledge bases; very cheap secondary storage; facilitation of artificial realities	Mid 1990's
High quality voice synthesis	Auditory natural language output	Late 1980's

User specific, limited vocabulary voice recognition	Restricted natural language input	Late 1990's
Sophisticated User Interface Management Systems	Easier development of programs; reduced time for novices to master an application	Early 1990's
Intelligent tutors and coaches for restricted domains	Models of embedded expertise for greater individualization, mastery	Year 2000+
Advanced manipulatory input devices	Mimetic usage which builds on real world experience	Early 1990's
Microworlds	Experience in applying theoretical information in practical situations	Late 1990's
Artificial Realities	Intensely motivating simulation and experience	Year 2000+
Cognitive audit trails	Support for finding patterns of suboptimal performance	Late 1980's
Consciousness sensors	Monitoring of mood, state of mind	Late 1990's
Intelligent, semi-autonomous agents	Support for user-defined independent actions	Mid 1990's
Computer-supported cooperative work (collaborative design, collective problem solving, group decision support), including WYSIWIS	Mastery of team task performance	Mid 1990's
Current mainframe performance on microcomputers	Sufficient power for advanced functionalities	Late 1990's

Knowledge processing and Knowledge Base Management Systems	Goal-oriented, context-specific access to concepts and skills	Late 1990's
Hypermedia	Interlinking of diverse subject matter; easier conceptual exploration, training, collaboration	Late 1980's
Synthesis of computers, telecommunications	Easy interconnection; realistic simulation	Early 1990's

Summary

The rapidity with which the advances above are developed will play a key role in determining the configuration and capabilities of TMIS and SSE. The immediate functional requirements set forth for both systems are on the cutting edge of available technology, and the projected complexity of SSP documentation needs will require the development of yet to be discovered hardware and software capabilities.

Current TMIS and SSE documentation requirements allude to an integrated computer system composed of workstations, mini-computers and mainframes in various geographic locations linked together through a high speed communications network. Full function word and graphics processing, file transfer, full text search and retrieval, electronic mail, displaying at least eight simultaneous colors, and search and retrieval of mixed object documents (text and graphics in a common file architecture) must be supported. These systems are expected to include optical scanners capable of processing "E" size engineering drawings and to have a standard, easy-to-use interface and a common command modality on workstations, minicomputers, or mainframes.

The current solution to such an architecture requires a multi-vendor combination of hardware, software and telecommunications, which will most likely not be well integrated in the short term. As the initial configurations of TMIS and SSE take shape, a conflict of requirements and costs is a probable outcome. In some cases, the cost of integrating current COTS to meet system requirements will be unacceptable, particularly where current trends as the OSI Standards activity will result in such integration in the mid-range future.

NASA's goal of technological leadership, coupled with the demanding time frame for SSP development, creates pressures to develop needed information technologies which are not currently available. While such an approach fulfills the goal of developing advanced spin-off technologies, it also consumes resources which are limited. A further complication may be the development of technology or standards based upon needs which do not conform to the direction of the rest of the information industry.

Trends in information system hardware may be capable of meeting TMIS and SSE information system requirements within projected SSP time frames. Questions that remain are 1) whether the availability of such capabilities are within the projected SSP budget range, 2) whether the requirements and timetable set forth within SSP implementation plans mesh with the evolutionary availability of these new generations of hardware, and 3) what the costs of early obsolescence of existing SSP information systems are likely to be.

STANDARDS, CONNECTIVITY, AND COMPATIBILITY

Since the early 1960's, the information technologies have exhibited exponential growth in capabilities, complexity, and applications. However, the large installed base of systems incompatible with each other poses a barrier to further rapid evolution. This problem, coupled with the need to make information technologies interconnect to facilitate the development of very large scale information systems, has prompted the creation of "standards" organizations.

The 1979 National Policy on Standards for the United States (NPS) described the essential concept of a standard as:

> A prescribed set of rules, conditions, or requirements concerning definition of terms; classification of components; specifications of materials, performance, or operations; delineation of procedures; or measurement of quality and quantity in describing materials, products, systems, services, or practices.

Traditionally, standards are intended to produce beneficial results; potential outcomes include [Cerni, 1984]:

- an increase in productivity and efficiency in industry because of larger scale, low-cost production of interchangeable, uniform parts
- a fostering of competition by allowing smaller firms to market products, readily acceptable by the consumer, without the need for a massive advertising budget
- greater dissemination of information; more technology transfer
- expanding international trade because of the feasible exchange of products among countries
- greater conservation of resources
- improvements in health and safety
- promotion of the world-wide exchange of information (voice and data)

Standards come in a confusing variety of shapes and sizes. There are object standards, documentary standards, and conceptual standards; all can be either voluntary or mandatory. The global information technology industry is shaped by fundamental standards (such as the use of the metric system), products standards or integrated standards which address the challenge of high technology systems operating in large networks (e.g. TMIS and SSE).

Historically, the development of standards was typically reactive; an environment in which a set of standards was needed exhibited a problem which served to trigger the development of a standards solution. In most cases, this reactive approach adequately suited the needs of the information industry in the sixties and early seventies. However, the rapid pace of technological evolution, increasing levels of invested resources, and the establishment of international participation in very large scale projects involving a high degree of information technology (such as space development) have prompted the need to develop a proactive approach to standardization. Large scale project complexity and an increasing awareness that an extended temporal duration can produce a corrosive effect to the systems which constitute the "institutional memory" of a complex effort have also contributed to the creation of standards designed to aid in managing and planning project evolution.

Nowhere have standards activities become more prevalent than in the two key areas of telecommunications and information processing, both directly related to the focus of this study. Standards, compatibility and connectivity are central goals of NASA's TMIS, SSE and AIM strategies previously discussed.

Before reviewing the current state of affairs of these standards areas, a brief discussion of the organizations responsible for the development of these standards will help to frame their use and impact in the NASA environment. The rationale of providing a summary of current global efforts for developing national and international standards for information processing and telecommunications is twofold:

1) TMIS, SSE, SSIS, the PSCN, and most NASA information technology are developed by private sector contractors, many of whom abide by the specifications of the standards set forth by the two major international standards bodies. A fundamental construct of the SSP information systems is that they be "off the shelf" technology wherever possible. Therefore it stands to reason that SSP information technology, particularly over an extended time frame, will be significantly influenced by standards.
2) The "lessons learned" from the creation of international standards may offer valuable insight in the development of SSP information system policies and strategies. NASA's AIM project encompasses a fairly complete subset of the organizations, efforts and obstacles faced by the standards-setting bodies. In a similar manner to those groups, NASA must integrate a wide variety of technologies in seeking to resolve highly complex problems which require the organized cooperation of a large number of organizations that are geographically and culturally different and are motivated by dissimilar economic and political incentives.

Standards Organizations

International standardization efforts are a kaleidoscope of names, companies, committees, and special interests, but two major standards bodies (ANSI and ISO) are

89

expected to have the greatest impact upon information technology (and indirectly SSP information systems) over the next fifteen years. The American National Standards Institute (ANSI) functions as the primary coordinator for planning, approving and maintaining National American Standards. ANSI is non-profit and is composed of over 200 non-profit participants (governments, trade associations, standards groups) and more than 1000 company members from the private sector. A major role played by ANSI is as the official U.S representative for two major non-governmental, international standards bodies: the International Standards Organization (ISO) and the International Electrotechnical Commission (IEC).

Two of the many ANSI Accredited Standards Committees (ASC) most relevant to this study are ASC X3, Information Processing Systems, and ASC T1, Telecommunications. ASC X3 is composed of approximately 80 technical subcommittees covering standardization of computers, information processing, and peripherals devices in such specific areas as:

- Recognition Systems
- Programming Languages
- Data Representation
- Systems Technology
- Media
- Documentation
- Communications

The scope of ASC T1 addresses the formulation of industry interconnection standards [Cerni, 1984]:

> Committee TI develops standards and technical reports related to interfaces for U.S. networks which form part of the North American telecommunications system. TI also develops positions on related subjects under consideration in various international standards bodies. Specifically, TI focuses on those functions and characteristics associated with the interconnection and interoperability of telecommunications networks at interfaces with end user systems, carriers, and information and enhanced service providers. These include switching, signaling, transmission, performance, operation, administration, and maintenance aspects. Committee TI is also concerned with procedural matters at points of interconnection, such as maintenance and provisioning methods and documentation, for which standardizations would benefit the telecommunication industry.

As of 1983, 24 technical subcommittees dealt with the following general areas:

- Carrier to Customer Premises Equipment Interfaces
- Integrated Services Digital Networks (ISDN)
- Internetwork Operations, Administration, Maintenance and Provisioning

- Performance
- Carrier to Carrier Interfaces
- Specialized Subjects

Three major international standards organizations also operate: the International Organization for Standardization (ISO), the International Electrotechnical Commission (IEC), and the International Telecommunication Union (ITU). Each of these organizations has technical committees which are devoted to standards issues similar to those considered by the ASC X3 and T1 committees. In the case of the ISO, its Information processing standards group is TC97; the IEC technical standards committee for Information Technology Equipment is TC83; and the ITU standards activities are carried out through the International Telegraph and Telephone Consultative Committee (CCITT), which functions as a permanent study committee of the ITU. The United States participates in all these organizations either through ANSI (for the ISO and IEC) or through an office of the State Department (U.S. CCITT) which represents U.S. interests within the ITU/CCITT organization.

Telecommunications Standards

Global efforts to develop voice and data services over a common set of digital network facilities are centralized in the concept of the Integrated Services Digital Network (ISDN). The driving force behind the creation of the ISDN was the CCITT, which saw the advantages of standardization given that the telephone network uses programmable digital switches and digital transmission paths to establish connections for different services.

During the early 1980's, the concept of the ISDN took a more defined configuration, as CCITT and participating members (particularly the United States) worked to forge standards. It became an accepted concept that several independent ISDN type networks would co-exist for a number of years. In speaking of the public networks' role in shaping the ISDN, Irwin Dorros, head of the technical division of Bell Communications Research Inc. stated [Cerni, 1984]:

> The most attractive feature of the ISDN concept to the existing public network exchange carriers is that it allows an evolution toward Information Age services without knowing what the demand mix of these services will be. Since it takes many years to evolve these large and ubiquitous networks to new capabilities by having a network target architecture that is robust to service forecast uncertainties, we can confidently invest in the future. If indeed all access to other networks and information services will be carried on a "digital pipe" of appropriate cross section, many mix of services will be accommodated in the mature ISDN era. Carriers will thereby ensure that they will not find themselves with the wrong capabilities 10 years from now.

There exists a remarkable similarity between 1) the concept and goals of the ISDN and (as later discussed) the OSI Reference Model and 2) the conceptual goals and objectives of TMIS, PSCN, and SSE.

Computing and Information Processing Standards

A parallel standards development effort in the field of computers and information processing emerged in the late 1970's with the introduction of the Open Systems Interconnection (OSI) Seven Layer Reference Model. Driven by the need to allow computing systems to communicate with each other regardless of make or model, this effort is contributing to the eventual merger between telecommunications and computing systems.

The ISO TC97 was primarily responsible for the development of the OSI model through its SC16 committee (established in 1977). In reference to the work of TC97/SC16, the committee that initiated the OSI studies, Day and Zimmerman stated [Cerni, 1984]:

> In most cases, the job of a standards committee is to take sets of commercial practices and the current research results when applicable and codify these procedures into a single standard that can be utilized by commercial products. SC16 was presented with a somewhat different problem: develop a set of standards which emerging products could converge to before the commercial practices were in place and while many of the more fundamental research problems remained unsolved. It would be presumptuous to say that SC16 solved this problem. They did, however, find a way to cope with the problem in such a way as to maximize flexibility and to minimize the impact of change brought on by new technologies or new techniques.

The charge of SC 16 was the development of a reference model which would provide the architecture for all future development of standards for worldwide, distributed information systems. This goal was initially achieved in 1983 with the approval of International Standard 7498 and CCITT Recommendation X.200, which specified the basic architecture of of the Open Systems Interconnection (OSI) model.

The OSI model defines [Kay and Powell, 1984]:

- the functions of each layer
- interlayer communications
- the protocols used for peer communication

As the name suggests, the Seven Layer Reference Model is composed of the following layered sub-systems:

1. *Physical Layer:* provides transparent transmission bit stream over a circuit built in some physical communications medium.
2. *Data Link Layer:* overcomes the limitations inherent in physical circuits and allows errors in transmission to be detected and recovered, thereby masking deficiencies in transmission quality.
3. *Network Layer:* transfers data transparently, selecting a route and directing the data accordingly.
4. *Transport Layer:* provides end-user to end-user transfer, optimizing the use of resources according to the type and character of the communication, and relieves the user of any concern for details of the transfer.
5. *Session Layer:* coordinates the interaction within each association between communicating application processes.
6. *Presentation Layer:* transforms the syntax of the data which is to be transferred into a form recognizable by the communicating application processes.
7. *Application Layer:* specifies the nature of the communication required to satisfy the user's needs. This is the highest layer in the Model and so does not have a boundary with a higher layer. The Application layer provides the sole means for application processes to access the OSI environment.

The ISO Seven Layer Model is the result of a decade of cooperative international effort and is presently the leading driving force on international standards for computing system architecture.

Summary

The ISDN and the OSI Model represent expansive global efforts to create proactive standards for telecommunications and computing systems. A substantial expenditure of resources is being directed at standards to which new systems could converge, rather than standardizing procedures after the fact.

It is important to note that both these efforts are in an early stage and, as such, the subject of intense competitive pressures from competing vendors. In particular the OSI Model, because of its complex nature and the far reaching impacts on computing technology, is the subject of competitive interests. The network architectures of both IBM and DEC are comprised of a seven layer approach which maps, somewhat differently, to the OSI Model; and both vendors are concerned forging the standards to their advantage.

The future of ISDN and OSI is gaining momentum as the standards become better defined, incorporated by greater numbers of developers, and are made a requirement in purchasing decisions by users. Through the early 1990's, computing and telecommunications systems increasingly comply with early versions of these standards.

Patchwork solutions may be prevalent, as non-compatible technology will need to be linked via the "black box" approach. However, by the mid-1990's, both standards will approach maturity in terms of definition, acceptance, and implementation; and global integration of computing and telecommunications will begin to become reality. A crucial benefit will be the ability to link distributed computing systems, thus simplifying the process of constructing very large scale information integrated telecommunications networks capable of transmitting virtually all forms of data at high rates of speed.

Such capabilities directly complement the requirements, goals and objectives of SSP information systems. The likely pervasiveness of these standards and their effects upon COTS imply that they should be considered as an important factor in NASA information system strategy. For example, whether the mature evolution of these capabilities will be available within the present SSP implementation timetable is questionable, thus raising the issue of the feasibility of current schedules. Also, to the extent that NASA chooses to participate in the process of creating these standards, functionalities important to the operation of TMIS and SSE can be incorporated. Overall, NASA's goal of technological leadership implies that a proactive stance on emerging standards is vital.

Organizational Impacts of Information Intensive Environments

The preceding sections have described both emerging functionalities for knowledge creation, capture, transfer, and utilization and the underlying hardware architectures and standards needed to achieve them. This section shifts focus: implementing widespread usage of documentation systems with these capabilities will create major changes in individual task performance, group dynamics, and management practices at NASA.

Anticipating the exact nature of these shifts and their organizational consequences will be difficult because information technologies are so flexible that they can be utilized in many ways to accomplish the same task. For example, an institution's word processing can be performed via central clerical pools, or through a distributed support staff integrated into non-clerical working groups, or by eliminating support staff with managers and professionals producing their own written materials [Johnson et al, 1985]. Each of those approaches will lead to very different effects on an organization's flow of information, work roles, salary schedule, need for skilled personnel, and staff morale.

Without knowing the eventual configuration and functionality of space station systems for knowledge creation, capture, transfer, and utilization, making specific predictions about their likely institutional effects is impossible. Even with foreknowledge of the properties of "next generation" SSE documentation systems, determining the consequences of implementing leading edge devices is challenging because no historical precedents exist from which to draw parallels.

The immediate institutional outcomes of technological innovation can also be quite different than their long range effects. Typically, new information technologies have their impact on organizations in four sequential stages [Coates, 1977]:

Stage One: The new technology is adopted by an institution to carry out existing functions more efficiently.

Stage Two: The institution changes internally (work roles, organizational structures) to take better advantage of these new efficiencies.

Stage Three: Institutions develop new functions and activities enabled by additional capabilities of the technology. As the roles of different types of institutions expand, new competitive relationships emerge.

Stage Four: The original role of the institution may become obsolete, be displaced, or be radically transformed as new goals dominate the institution's activities.

This sequential evolution adds to the complexity of forecasting organizational impacts. For example, computer-aided design (CAD) was originally conceived as a more efficient way of producing engineering drawings. Now, the capabilities of this task performance tool have transformed design processes and are driving the integration of production across design, development, and manufacturing.

Despite these difficulties in predicting consequences, anticipating individual and institutional outcomes is vital to building user acceptance and to tailoring system design for optimal productivity. Under conditions of uncertainty, the best strategy is 1) to search for general heuristics which seem useful in understanding an organization's transition to an information intensive environment and 2) to introduce changes in documentation systems in a carefully planned, but flexible manner.

Implementing a new information system is a complex strategic intervention, affecting the social organization of work and patterns of institutional control. For example, desktop computing can alter several dimensions of occupational roles [Kling & Iacono, 1987]:

content: level of specialization, complexity, abstractness
social contact: face-to-face, working alone, machine-mediated
structure: pace, pay, career opportunities, work standards, required skills, power, control over working conditions, status
orientation: perceived individual and institutional mission
time: system utilization, training for use

Even after the new system is established, continuous review of the social architecture of the computer-based work setting throughout its life cycle is important.

Consequences of Implementing Advanced Documentation Systems

Listed below are probable "side effects" of an organization becoming more information intensive by using advanced, integrated documentation technologies. The list is illustrative rather than complete; each heuristic is summarized very briefly, in no particular order. For reasons of space, the rationale for each is not discussed, but the citations will provide ample detail if consulted. None of these outcomes will necessarily apply to NASA; as discussed above, how new systems for knowledge creation, capture, transfer, and utilization are implemented will determine which of these general tendencies are actually realized in space station development.

- *More of the institution's production may be achieved by coordinating external suppliers, contractors, and vendors; less by the activities of internal personnel.* Malone, Yates, and Benjamin [1987] distinguish between electronic markets (which coordinate the flow of materials or services through supply and demand forces and through external transactions among different individuals and firms) and electronic hierarchies (which coordinate product flow through adjacent steps by controlling and directing activities at higher levels in a managerial hierarchy). Contracting out the creation of a software module with multiple bidders competing would be a market model of production; competitive forces determine the design, price, quantity, and target schedule. Using a sole source (internal or external) would be hierarchical; design, price, quantity, and delivery schedules are determined by managerial decisions.

Advanced information systems empower the management of both markets and hierarchies, but the costs of market coordination are more greatly reduced than those of hierarchical coordination, leading to a potential shift more toward market activities. In general, organizational hierarchies based on power, control, secrecy, ownership, access, and geography may erode due to advanced communication networks [Cleveland, 1985].

- *Mainframe computers may be replaced for many applications by alternative organizational structures for computing*. Malone and Smith [1986] have developed a model for comparing the productive efficiency of different types of computer system architectures in achieving institutional coordination. Depending on reliability and cost considerations, separate personal computers, networks of shared processors with centralized scheduling, or networks of shared processors with decentralized scheduling are preferable to mainframes.
- *Intensive computer conferencing may lead to a decrease in middle level management functions, with an increase in the complexity of staff specialist and site management roles*. Crowston, Malone, and Yin [1987] conducted a case study in which message content was analyzed as a computer conferencing system was introduced. The collecting and filtering roles of middle managers were eroded by the increased access to information that electronic messaging provided. Simultaneously, universal and inexpensive communication enabled greater job differentiation and functional specialization higher up in the organizational hierarchy. Other types of role changes from advanced information systems may occur as well, depending on implementation strategy [Chamot, 1987].
- *Issues of individual privacy, data security, and system reliability will become more important in an organization using electronic information systems*. The usage of networked workstations permits a variety of new measures of individual productivity, including continuous monitoring of employee

performance; this raises issues of privacy and morale. Large scale, integrated knowledge bases carry the risk of increased vulnerability to improper access into system functions or sensitive data. When an advanced documentation system has performance difficulties, a severe impairment of organizational effectiveness can result. All these illustrate that powerful systems for knowledge creation, capture, transfer, and utilization can be a "double edged sword" unless designed and managed carefully [Office of Technology Assessment, 1981].

- *The complex coordination enabled by advanced information systems may produce a qualitative increase in the collective intelligence of an institution.* Sometimes, an organization behaves foolishly when confronted with problems even though its employees individually recognize that those collective actions are stupid. Situations in which the behavior of an institution is less than the sum of its parts are all too common. However, Minsky [1986] argues that intelligent agents, when properly organized, can form a "superintelligence" for task accomplishment. As a greater understanding of knowledge creation, capture, transfer, and utilization systems evolves, a more powerful, systemic organizational intelligence may emerge. Similarly, institutional performance may increase as corporate memory becomes more powerful and accessible through advanced knowledge systems.
- *Rewarding employee merit through salary increases or increased supervisory responsibilities is becoming more difficult for many institutions, and using advanced information tools as an alternative form of job enhancement may become more prevalent.* Employment quality is characterized by a number of measures, especially mental challenge. Worker satisfaction and job commitment rise when systems are introduced which increase opportunities for learning, creativity, autonomy, responsibility, variety, and coping with diverse difficulties [Kraut, 1987a]. The enhanced social status users of advanced technology enjoy is also a potential reward [Straussman, 1985].
- *Fragmenting the process of programming through overautomation and excessive reliance on assembly-line metaphors can deskill work and produce job dissatisfaction.* Kraft [1987] documents how imposing industrial models on the software coding process can have adverse effects on programmers. In general, fragmenting work and creating narrowly specialized occupational roles can produce long term organizational problems, even though short term efficiencies may result.

- *Desirable employee attributes may include an increased emphasis on higher order cognitive skills (creativity, complex pattern recognition, systemic thinking) and on affective skills of cooperation, compromise, and group decision making*. Partnerships with intelligent tools may change individual learning and thinking styles (as discussed in the **Advanced User Interface Capabilities** section); and the motivational and emotional characteristics necessary for flexible, collective task performance are likely to become more important [Dede, 1987c]. As expert systems automate many routine tasks involved with space station development, employee training may also change [Dede, 1985].

While incomplete and general, this illustrative list of potential effects indicates the range of issues which need to be considered in the design and utilization of advanced documentation systems. Overall, the computer is an "evocative object" which tends to alter intellectual, emotional, and motivational attributes of its users. While most of the personal consequences of advanced information systems are positive, care must be taken to avoid inappropriate fascination with the machine [Turkle, 1984].

Heuristics for Improving Implementation

Using change management strategies can be important even if no transition problems are expected and users are technologically sophisticated. For example, telework (the use of information technologies to accomplish tasks in settings remote from an office environment) has developed far more slowly than predicted because of unanticipated psychological and social barriers. People come to work in offices for a variety of personal reasons, not just because that's where their desk is located [Kraut, 1987b].

A review of the considerable literature on approaches to organizational innovation and change management is beyond the scope of this report. An overview and comparative analysis of office automation methodologies and implementation strategies are presented in Hirschheim [1985]. A few ideas are discussed below to illustrate the types of policies which can aid in implementing electronic systems for knowledge creation, capture, transfer, and utilization at NASA.

On the basis of large scale empirical analysis of information system end users, Danzinger and Kraemer [1986] identify five conditions crucial to productive utilization:

1) decentralized computer packages in the hands of users,
2) applications which enable facile human-machine interaction,
3) user competency and experience with computers,
4) support personnel sensitive to users' felt needs for advice, and
5) routine rather than selective usage of information systems.

Other factors which correlate with organizational success in introducing new information technologies include [Bikson, 1987]:

- a balanced approach between social and technical aspects of innovation,
- extensive support for users learning the new system,
- a positive institutional orientation toward change,
- user participation in the decision process, and
- rewarding workers who facilitate diffusing the new innovation through the institution.

To illustrate a more detailed discussion of one of these factors, user acceptance of new information systems can be enhanced by multiple sources of human help in learning and using the system. (Self-explicating interfaces, as discussed in the section on **Advanced User Interface Capabilities**, will take time to develop.) Strategies by which users can help each other include both direct interactions with co-workers who are designated consultants ("local experts") and machine-mediated exchanges (through hotlines, computer conferencing, and electronic bulletin boards) [Bannon, 1987]. On a more general level, Blomberg [1987] describes how the social interactional environment of organizations can be tailored to facilitate user understanding and acceptance. For example, steps to reduce user anonymity can be useful in early diagnosis and remediation of system problems, as well as increasing user expertise.

In general, technological innovation strategies fall into three categories [Taylor, 1987]:

1) *Problem-oriented* approaches to organizational improvement and effectiveness center on what has gone wrong historically (past-focused).
2) *Solution-oriented* approaches examine new external developments to find ways to improve current functioning (present-focused).
3) *Purpose-oriented* approaches pose a fundamental reevaluation of organizational mission and goals based on emerging technological capabilities (future-focused).

Of the three, the purpose-oriented perspective--which involves reinventing the uses of a technology--is often the most neglected in the innovation process [Johnson & Rice, 1987].

Research indicates that decision makers remote from system users and their managerial hierarchy frequently provide resources, income, legitimacy, or staff vital to the productive operation of the knowledge base. Often, many outcomes of a new information system are unexpected because significant participants in shaping its adoption or use are mistakenly excluded from traditional impact assessments. Kling

[1987] delineates how "web" models based on resource-dependency relationships can improve the anticipation of implementation problems and can aid in their resolution.

Summary

The two most common errors in technology assessment are overestimating the speed of diffusion of an innovation and underestimating its eventual consequences. The extent to which individual and collective resistance to change can retard the utilization of new technologies should not be underestimated. While every other section of this study focuses on technical and cognitive challenges in the evolution of electronic documentation, psychological and social barriers may be the most profound obstacles faced in implementing advanced information systems.

Isaac Asimov once said that the important thing to forecast is not the television, but the soap opera; not the automobile, but the parking problem. Similarly, the long term, higher order outcomes of an information technology may be more profound than the effects for which it was originally implemented. The impact of the medium can be more important than that of the message [Winograd & Flores, 1986].

High Leverage Topics for Further Exploration

What should be NASA's response to the trends and developments summarized in this study? The research directions discussed have the potential to reshape information systems in fundamental ways, but these new approaches are still in the prototype stage and far from the robust, dependable functioning needed for implementation. In some cases, whether remaining technical and conceptual obstacles can be resolved is an open question. However, each of these concepts is sufficiently promising that its emergence into common practice in the next decade seems plausible.

As with other institutions, NASA's goals sometimes conflict with each other. To ensure timely, safe, low cost, and effective space station development, a conservative stance toward relatively untried approaches is often warranted. As a result, in its early stages SSP frequently utilizes "off the shelf" familiar applications that may be less effective than emerging, but possibly undependable techniques.

On the other hand, another basic goal of the U.S. space program is the creation of advanced processes for very large system design and development. Spin-offs with the potential for American economic leadership can only come when NASA applies leading edge strategies and invents its own solutions to problems. Thus, the evolution of the space station requires a delicate balance between innovation and conservatism in adopting new techniques.

The section on **Standards, Compatibility, and Connectivity** presents several reasons why NASA's involvement in shaping new technical developments is important. Essential in accomplishing this goal is the sponsorship of substantial research focused on space station issues; tailored to mesh with existing SSP applications; designed to enhance internal expertise in advanced concepts; and targeted to result in the development of practical, robust information systems. NASA's current activities in sponsoring such research indicate a strong level of internal commitment to this type of exploration. Hopefully, this research agenda will continue to be of high priority despite a troubled funding situation for the space station project.

To follow up on the themes discussed in this study, an effective strategy would be to sponsor clusters of sequenced projects; each cluster would test the applicability to SSP of advanced ideas in knowledge creation, capture, transfer, and utilization. Early projects in each sequence would be small-scale and targeted more to exploratory research than the production of a practical information system. Multiple, parallel efforts would be funded, each focused on a single innovation within that theme (e.g. hypermedia applications to software documentation).

Later endeavors in the sequence would be larger in scope; would integrate several parallel projects (e.g. a hypermedia-based training system with a direct manipulation interface and group decision support capabilities); would be linked to existing NASA applications; and would be directed toward a robust, usable product. The early stages of concept development could be omitted for some topics which are already relatively well understood (e.g. mimetic interfaces).

Below are presented potential research clusters which build on the developments and advances discussed in this report. This list is limited in several ways:

- The themes delineated are tailored around the paradigmatic example of the SSE. Comparable work could be initiated to explore applications for TMIS, SSIS, and PSCN. Also, generic research could be done on issues common to all SSP information systems.
- The perspective underlying these suggestions is "electronic documentation." Alternative orientations (such as "configuration management and control") would result in different types of projects.
- Resources regional to the Johnson Space Center, which sponsored this report, are highlighted.
- The list is deliberately kept short, at the risk of omitting many projects of potential value. Long collections of suggested activities present problems of overchoice; also, recent Congressional actions cutting back on SSP funding have drastically restricted the resources available for this type of research.

Thus, the list is intended to be suggestive rather than definitive, and readers are encouraged to generate their own additional ideas for projects.

Illustrative Research Projects

As discussed in this report's **Introduction**, a common, overarching goal for improving SSP documentation systems would be the creation of a technology-based "knowledge structure" which unobtrusively amplifies human cognition and memory. Each research theme below exemplifies one aspect of that goal as applied to the SSE. The initial projects are listed in no particular order.

OVERARCHING ACTIVITIES

» *A User-Based, Detailed Description of a Future SSE Workstation*

This project would identify major types of SSE users (e.g. managers, programmers) and select representative "power users" from each category. This group would be given a rich background in knowledge creation, capture, transfer, and utilization from a software engineering perspective. They would then construct, integrating their different perspectives, a detailed description of the functionalities and capabilities an SSE workstation should embody by the Evolutionary Phase (after the station is operational). A critical path analysis would be conducted, delineating the advances (computational, conceptual) needed for developing such a workstation. A forecast of the estimated availability and costs of these capabilities would also be prepared.

(Such a project would be a follow-on activity to this study, targeting the focus to SSE, deepening the level of detail, and involving a variety of NASA personnel.)

» *A Compilation of NASA's Internal SSE Research on Knowledge Bases*

The spectrum of current NASA SSE-related projects would be examined for research which overlaps with the themes identified in this report's external scan. Those NASA researchers would be given briefings on the overarching context in knowledge creation, capture, transfer, and utilization. An electronic network would be established linking NASA researchers with external groups identified in this study who are pursuing topics of common interest. Joint workshops targeted to crucial issues would be funded.

(This would both extend the groups involved in research of interest to SSE and serve as a dissemination mechanism for the ideas in this report.)

THEMATIC EXPLORATIONS

» *Constructing a Prototype Hypertext SSE Documentation System*

Scacchi's Document Integration Facility (DIF) and Tektronix's Neptune are potential models for applying hypertext to software documentation. The Hypertext Group in the Department of Computer Science at Texas A&M University, under the directorship of Dr. John Leggett, is a strong regional resource for the Johnson Space Center to use in building such a prototype. These researchers are already engaged in constructing hypertext-based documentation for IBM and have good ties with the Microelectronics and Computer Technology Consortium's (MCC) second generation hypertext project, which is another regional resource. Such a prototype system would test the utility of hypertext in improving group software design collaboration, automating SSE code documentation, recording detailed rationales for SSE design decisions, reducing redundancy in record keeping, and enforcing consistency when SSE documentation is altered.

(This research would aid in identifying issues unique to SSP in using hypertext for software documentation. For example, current implementations such as DIF and Neptune are not directed toward the Ada environment.)

» *Building a simulated SSE interface with advanced capabilities*

Researchers familiar with NASA's work (such as Shneiderman, the Human Factors group at Ames, Nickerson's team at BB&N, Tullis, and Rudisill) could be involved in the design of a "dummy" model which would simulate the types of functionalities described in the **Advanced User Interface Capabilities** section.

(Users often find that abstractly conceptualizing the implications of new cognitive environments for their task performance is very difficult. Having a shell "advanced interface" to demonstrate the capabilities of an sophisticated SSE workstation would aid in resolving this problem.)

» *Testing Knowledge Transfer Techniques' Effectiveness for Training*

Hypertext documentation, mimetic interfaces, intelligent coaches, microworlds, and similar advanced functionalities are expected to speed novices' ability to master sophisticated task performance tools. This project would develop small-scale training prototypes using these capabilities and evaluate their effectiveness compared with traditional SSE instruction.

(Among the highest leverage outcomes of improved documentation are reduced training and apprenticeship costs and increased user productivity.)

» *Developing Collaborative Design Capabilities for the SSE*

A prototype, hypertext-based, collaborative design system would be developed and tested with a representative group of SSE personnel. The MCC's Leonardo project is a regional resource on software design environments with hypertext and CSCW components.

(Surfacing collective assumptions, resolving conflicts, and developing shared models of a task are vital to the success of SSE. This research would aid in identifying CSCW implementation challenges unique to SSP, such as the usage of Ada.)

» *Synthesizing Research on the Management of Knowledge Systems*

Different sections of this study have identified emerging strategies useful in managing massive, complex, long term engineering projects. These approaches are being developed from very different perspectives--CSCW aids in group decision support; ethnographic analyses of how organizations change when advanced information tools are implemented; knowledge based project management systems--and are not well integrated. This project would review and synthesize management-related research targeted to software development environments similar to SSE. Applications to current SSP policies, practices, and plans would be delineated.

(Social and political limits, rather than technical constraints, will determine the speed of SSE evolution, so the usage of leading edge management strategies is vital.)

» Synthesizing Research on Software Engineering

As discussed in the **Introduction**, numerous projects are exploring advanced functionalities for improving the software development life cycle. A scan and synthesis of these emerging capabilities comparable in its scope to this study would be prepared, including a forecast of probable availability and cost. (A survey of these specific SSE functionalities would complement the more generic knowledge base perspective of this study.)

Overall, using such projects to skew the development of SSE toward knowledge creation, capture, transfer, and utilization perspectives may result in the following benefits:

- augmenting individual creativity and productivity
- improving group software design collaboration
- enhancing the speed and quality of SSE coding
- automating SSE code documentation
- recording detailed rationales for SSE design decisions
- reducing redundancy in record keeping
- speeding the entry of new information
- enforcing consistency when SSE documentation is altered
- improving efficiency and effectiveness of SSE training
- increasing rapidity of access to SSE documentation materials
- linking SSE to other SSP documentation systems more readily
- improving group decision support
- facilitating collaboration-at-a-distance

Conclusion

One purpose of this report is to interest experts external to NASA in using the space station as a very challenging testbed for innovative information system techniques. Another purpose is to link these external resources with space station researchers internal to NASA, its contractors, and its grantees. While a catalog of related NASA projects was not an objective of this study (and therefore relevant internal work may not have been cited), the numerous references to research conducted or sponsored by NASA indicate the high level of interest already focused on the themes in this report.

Timely initiation of a research agenda targeted to these topics seems crucial to the successful evolution of SSP information systems. Already, decisions being made about the initial configuration of SSE and TMIS have the potential to foreclose future options which could greatly increase these systems' utility. Retrofitting emerging approaches to information systems is very difficult unless their designers build in sufficient flexibility for an orderly progression of increasingly sophisticated strategies. Research-based prototypes which demonstrate the feasibility and value of the advanced concepts discussed in this report are an essential step toward such a design strategy.

References

Abbott, Russell J. (1987) Knowledge Abstraction. *Communications of the ACM* **30**, 8 (August, 1987), 664-671.

Acksyn, Robert, McCracken, Donald, and Yoder, Elise. (1987) KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations. Proceedings of Hypertext '87. Chapel Hill, NC: University of North Carolina.

Anderson, John R. (1983) The Architecture of Cognition. Cambridge, MA: Harvard University Press.

Anderson, J.R., Farrell, R., and Sauers, R. (1984) Learning to Program in LISP. *Cognitive Science* **8**, 87-129.

Bannon, Liam J. (1987) Helping Users Help Each Other. In Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Bateson, Gregory. (1975) Steps to an Ecology of Mind. New York: Ballantine.

Bezold, C., and Olson, R. (1986) The Information Millennium: Alternative Futures. Alexandria, VA: Institute For Alternative Futures.

Bikson, Tora K. (1987) Understanding the Implementation of Office Technology. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Blair, D.C. and Maron, M.E. (1985) An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System. *Communications of the ACM* **28**, 3 (March, 1985), 289-299.

Blomberg, Jeanette L. (1987) Social Interaction and Office Communication: Effects on User Evaluation of New Technologies. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Bolt, R.A. (1979) Spatial Data Management. Cambridge, MA: MIT Press.

Borning, A. (1981) The Programming Language Aspects of ThingLab, a Constraint Oriented Simulation Laboratory. *ACM Transactions on Programming Languages and Systems* **3**, 4, 353-387.

Brachman, Ronald J., and Levesque, Hector J., Eds. (1985) Readings in Knowledge Representation. Los Altos, CA: Morgan Kaufmann.

Brachman, Ronald J., and Levesque, Hector J. (1986) What Makes a Knowledge Base Knowledgable? Larry Kerschberg, Ed., Expert Database Systems: Proceedings From the First International Workshop. Menlo Park, CA: Benjamin/Cummings Publishing.

Brodie, Michael L. (1986) Large-Scale Knowledge Based Systems: Concluding Remarks and Technological Challenges. Michael Brodie & John Mylopoulos, Eds., On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies. New York: Springer-Verlag.

Brodie, Michael L. and Mylopoulos, John (Eds.). (1986) On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies. New York: Springer-Verlag.

Brown, John Seely. (1985) Process Versus Product: A Perspective on Tools for Communal and Informal Electronic Learning. *Journal of Educational Computing Research* 1 (1985), 179-202.

Brown, John Seely, and Moskovitz, Gary. (1985) AI: Windows of Opportunity in Office Automation. Palo Alto, CA: Xerox PARC.

Brown, John Seely. (1986) From Cognitive to Social Ergonomics. In Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Burton, Richard R. (1986) The Environment Module of Intelligent Tutoring Systems. Palo Alto, CA: Xerox Palo Alto Research Center.

Campbell, William J. and Roelofs, Larry H. (1986) The Development of an Intelligent User Interface for NASA's Scientific Databases. *Telematics and Infomatics* **3**, 3, 177-190.

Card, S.K., Moran, T.P., and Newell, A. (1983) The Psychology of Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Card, Stuart K., and Henderson, D. Austin. (1987) A Multiple, Virtual-Workspace Interface to Support User Task Switching. Proceedings of CHI+GI 1987. New York: ACM.

Carroll, John M. and McKendree, Jean. (1987) Interface Design Issues for Advice-Giving Expert Systems. *Communications of the ACM* **30**, 1 (January, 1987), 14-31.

Carroll, John M., and Rosson, Mary Beth. (1987) Paradox of the Active User. John M. Carroll (Ed.), Interfacing Thought: Cognitive Aspects of Human-Computer Interaction. Cambridge, MA: MIT Press.

Cerni, D.M. (1984) Standards in Process: Foundations and Profiles of ISDN and OSI Studies. Washington, DC: National Telecommunications and Information Administration, U.S. Department of Commerce.

Chamot, Dennis. (1987) Electronic Work and the White-Collar Employee. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Clancey, William J. (1987) Knowledge-Based Tutoring: The Guidon Program. Cambridge, MA: MIT Press.

Cleveland, Harlan. (1985) The Twilight of Hierarchy. In Bruce R. Guile (Ed.), Information Technologies and Social Transformation. Washington, DC: National Academy of Engineering.

Coates, Joseph F. (1977) Aspects of Innovation: Public Policy Issues in Telecommunications Development. *Telecommunications Policy* **1**, 3, 11-23.

Conklin, Jeff, and Richter, Charles. (1985) Support for Exploratory Design. Austin, TX: MCC Technical Report STP-117-85, Microelectronics and Computer Technology Consortium.

Conklin, Jeff. (1986a) A Survey of Hypertext. Austin, TX: MCC Technical Report STP-356-86, Microelectronics and Computer Technology Consortium.

Conklin, Jeff. (1986b) A Theory and Tool for Coordination of Design Conversations. Austin, TX: Technical Report STP-236-86, MCC.

Conklin, Jeff. (1987) Hypertext: An Introduction and Survey. *Computer* **20**, 9, 17-41.

Corkill, D.D., and Lesser, V.R. (1983) The Use of Meta-Level Control for Coordination in A Distributed Problem Solving Network. Proceedings of the International Joint Conference on Artificial Intelligence. Los Altos, CA: Morgan Kauffman.

Crowston, Kevin, Malone, Thomas W., and Lin, Felix. (1987) Cognitive Science and Organizational Design: A Case Study of Computer Conferencing. *Human-Computer Interaction* (in press).

Cypher, Allen. (1987) The Structure of Users' Activities. Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Danzinger, James N. and Kraemer, Kenneth L. (1986) People and Computers: The Impacts of Computing on End Users in Organizations. New York: Columbia University Press.

Dawkins, Richard. (1976) The Selfish Gene. New York: Oxford University Press.

Dede, Christopher J. (1985) An Alternative Paradigm for Space Station Training Based on Artificial Intelligence. Houston, TX: Training Division, Johnson Space Center.

Dede, Christopher J. (1986) A Review and Synthesis of Recent Research in Intelligent Computer-Assisted Instruction. *International Journal of Man-Machine Studies* **24**, 4, 329-353.

Dede, Christopher J. (1987a) Artificial Intelligence Applications to High Technology Training. *Journal of Educational Communications and Technology* (in press).

Dede, Christopher J. (1987b) Empowering Environments, Hypermedia, and Microworlds. *The Computing Teacher* **15**, 3 (November, 1987), 20-25.

Dede, Christopher J. (1987c) Implementation of Artificial Intelligence in Education: Two Scenarios. Austin, TX: Center for Research on Communication Technology and Society, University of Texas.

Dede, Christopher J., and Swigger, Kathleen M. (1988) The Evolution of Instructional Design Principles for Intelligent Computer Assisted Instruction. *Journal of Instructional Design* (in press).

Delisle, N. and Schwartz, M. (1986a) Neptune: a Hypertext System for CAD Applications. Proceedings of SIGMOD 1986. New York: ACM.

Delisle, N. and Schwartz, M. (1986b) Contexts: A Partitioning Concept for Hypertext. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

diSessa, Andrea A. (1985) A Principled Design for an Integrated Computational Environment. *Human-Computer Interaction* **1**, 1, 1-57.

diSessa, Andrea A. (1986a) Notes on the Future of Programming: Breaking the Utility Barrier. In Donald A. Norman and Stephen W. Draper (Eds.), User Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

diSessa, Andrea A. (1986b) Models of Computation. In Donald A. Norman and Stephen W. Draper (Eds.), User Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Donahue, James, and Widom, Jennifer. (1986) Whiteboards: A Graphical Database Tool. *ACM Transactions on Office Information Systems* **4**, 1, 24-41.

Dongarra, Jack C., Ed. (1987) Experimental Parallel Computing Architectures. Amsterdam: Elsevier Science Publishing Company, Inc.

Drexler, K. Eric. (1986) Engines of Creation. New York: Doubleday.

Evans, Michael W., and Marciniak, John J. (1987) Software Quality Assurance and Management. New York: Wiley & Sons.

Fairchild, Kim F. (1984) Computer Generated Synthesized Imagery. Minneapolis, MN: Computer Science Department, University of Minnesota.

Fairchild, Kim F. and Gullichsen, Eric. (1986) From Modern Alchemy to a New Renaissance. Austin, TX: MCC Technical Report HI-400-86, Microelectronics and Computer Technology Corporation.

Fairchild, Kim F., Poltrock, S.E., and Furnas, G.W. (1987) SemNet: Three-Dimensional Graphic Representations of Large Knowledge Bases. R. Guindon (Ed.), Cognitive Science and its Applications for Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Foley, James D. (1987) Interfaces for Advanced Computing. *Scientific American* **257**,4 (October, 1987), 126-135.

Fox, Edward A. (1986) Information Retrieval: Research into New Capabilities. S. Lambert and S. Ropiequet, Eds., CD ROM: The New Papyrus. Redmond, WA: Microsoft Press.

Frenkel, Karen A. (1985) Toward Automating the Software Development Cycle. *Communications of the ACM* **28**, 6 (June, 1985), 578-589.

Friedland, Peter. (1987) Knowledge Servers -- Applications of Artificial Intelligence to Advanced Space Information Systems. Moffett Field, CA: Artificial Intelligence Research Branch, NASA Ames Research Center.

Garg, Pankaj K., Jazzar, Abdulaziz, and Scacchi, Walt. (1986a) System Documentation as a Medium for Cooperative Work. Los Angeles, CA: Department of Computer Science, University of Southern California.

Garg, Pankaj K., Jazzar, Abdulaziz, and Scacchi, Walt. (1986b) DIF: A Document Integration Facility. Los Angeles, CA: Computer Science Department, University of Southern California.

Garrett, L. Nancy, Smith, Karen, and Meyrowitz, Norman. (1986) Intermedia: Issues, Strategies, and Tactics in the Design of a Hypermedia Document System. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Gentner, D., and Stevens, A. (1983) Mental Models. Hillsdale, NJ: Lawrence Erlbaum.

Georgeff, Michael P. (1987) An Embedded Real-Time Reasoning System. Menlo Park, CA: Artificial Intelligence Center, SRI International.

Gibson, William. (1984) Neuromancer. New York: Ace.

Goldstein, I. P. and Bobrow, D.G. (1984) A Layered Approach to Software Design. D. Barstow, H. Shrobe, and E. Sandewall (Eds.). Interactive Programming Environments. New York: McGraw-Hill.

Gould, John D., Alfaro, Lizette, Finn, Rich, Haupt, Brian, Minuto, Angela, and Salaun, Josiane. (1987) Why Reading was Slower from CRT Displays than Paper. Proceedings of CHI+GI 1987. New York: ACM.

Gullichsen, E., D'Souza, D., Lincoln, P., and Casey, T. (1986) The PlaneText Book. Austin, TX: MCC TR No. STP-333-86 (P), Microelectronics and Computer Technology Consortium.

Halasz, Frank G., Moran, Thomas P., and Trigg, Randall H. (1987) NoteCards in a Nutshell. Proceedings of CHI+GI 1987. New York: ACM.

Halasz, Frank G. (1987) Reflections on NoteCards: Seven Issues for the Next Generation of Hypertext Systems. Proceedings of Hypertext '87. Chapel Hill, NC: University of North Carolina.

Hall, Edward T. (1973) The Silent Language. New York: Doubleday.

Harlan, C.S. (1986) TMIS Overview Briefing. Houston, TX: Johnson Space Center, 1986.

Hillis, W. Daniel, and Steele, Guy L., Jr. (1986) Data Parallel Algorithms. *Communications of the ACM* **29**, 12 (December, 1986), 1170-1183.

Hillis, W. Daniel. (1987) The Connection Machine. *Scientific American* **256**, 6 (June, 1987), 108-115.

Hirschheim, R.A. (1985) Office Automation: A Social and Organizational Perspective. New York: Wiley.

Hopfield, J.J., and Tank, D.W. (1986) Computing with Neural Circuits: A Model. *Science* **233**, 4764 (August 8, 1986), 625-633.

Hutcheson, L.D. (1986) Design Criteria For AlGaAs/GaAs Integrated Optoelectronic Designs. Integrated Optical Circuit Engineering IV. Cambridge, MA: The International Society for Optical Engineering.

Hutchins, Edwin L., Hollan, James D., and Norman, Donald A. (1987) Direct Manipulation Interfaces. Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Johnson, B., Eveland, J., Taylor, J., Rice, R., Cline, T., Smith, D., and Browning, L. (1985) Innovation in Office Systems Implementation. Washington, DC: National Science Foundation.

Johnson, B. and Rice, R. (1987) Managing Organizational Innovation: The Evolution from Word Processing to Office Information Systems. New York: Columbia University Press.

Johnson, Lewis. (1987) Intention-Based Diagnosis of Errors in Novice Programs. Los Altos: Morgan Kauffman.

Jonassen, David H. (1986) Hypertext Principles for Text and Courseware Design. *Educational Psychologist* **21**, 4 (Fall, 1986), 269-292.

Kahn, Robert E. (1987) Networks for Advanced Computing. *Scientific American* 257, 4 (October, 1987), 136-143.

Kay, P. and Powell, P., Eds. (1984) Future Information Technology - Telecommunications. Washington DC: U.S. Department of Commerce.

Kerschberg, Larry. (1986) Opening Preface. Larry Kerschberg, Ed., Expert Database Systems: Proceedings From the First International Workshop. Menlo Park, CA: Benjamin/Cummings Publishing.

Kimura, Gary D. (1986) A Structure Editor for Abstract Document Objects. *IEEE Transactions on Software Engineering* SE-12, 3 (March, 1986), 417-435.

Kling, Rob and Iacono, Suzanne. (1987) Desktop Computerization and Work in Future Offices. In Richard Gordon and Linda Kimball (Eds.), Automation, Employment, and Work. Berkeley, CA: University of California Press.

Kling, Rob. (1987) Defining the Boundaries of Computing Across Complex Organizations. In R. Boland and R. Hirschheim (Eds.), Critical Issues in Information Systems. New York: John Wiley.

Knowledge Based Systems Laboratory. (1987) Knowledge Based Integrated Information Systems Development Plan, Volumes I and II. College Station, TX: Department of Industrial Engineering, Texas A&M.

Kraft, Philip. (1987) Computers and the Automation of Work. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Kraemer, Kenneth L., and King, John Leslie. (1986) Computer-Based Systems for Cooperative Work and Group Decision Making: Status of Use and Problems in Development. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Kraut, Robert E., Galegher, Jolene, and Egido, Carmen. (1986) Relationships and Tasks in Scientific Research Collaboration. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Kraut, Robert E. (1987a) Social Issues and White-Collar Technology: An Overview. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Kraut, Robert E. (1987b) Predicting the Use of Technology: The Case of Telework. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Lantz, Keith A. (1986) An Experiment in Integrated MultiMedia Conferencing. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Larson, Anna C. and Rienzo, John C. (1987) Computer Based Ada Training System. Middletown, RI: Softech Technical Briefing Document.

Laurel, Brenda K. (1986) Interface as Mimesis. Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Lenat, Doug, Prakash, Mayank, and Shepherd, Mary. (1986) CYC: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Acquisition Bottlenecks. *AI Magazine* **6**, 4, 65-85.

Lesk, Michael. (1984) Computer Software for Information Management. *Scientific American* **251**, 3 (September, 1984), 162-172.

Lewis, Matthew W., Milson, Robert, and Anderson, John R. (1987) The TEACHER'S APPRENTICE: Designing an Intelligent Authoring System for High School Mathematics. Greg P. Kearsley (Ed.), Artificial Intelligence and Instruction: Applications and Methods. New York: Addison-Wesley.

Lippman, A. (1984) Imaging and Interactivity. Proceedings of the Fifteenth Joint Conference on Image Technology. New York: ACM.

Loftin, R. Bowen, Wang, Lui, Baffes, Paul, and Rua, Monica. (1988) An Intelligent Training System for Space Shuttle Flight Controllers. Houston, TX: Artificial Intelligence Section, Johnson Space Center.

Lowe, David G. (1985) Cooperative Structuring of Information: The Representation of Reasoning and Debate. *International Journal of Man-Machine Studies* **23**, 97-111.

Malone, Thomas W. and Lepper, Mark R. (1985) Making Learning Fun: A Taxonomy of Intrinsic Motivations for Learning. R.E. Snow and M.J. Farr (Eds.), Aptitude, Learning, and Instruction: III. Conative and Affective Process Analysis. Hillsdale, NJ: Lawrence Erlbaum.

Malone, Thomas W. and Smith, Stephen A. (1986) Modeling the Performance of Organizational Structures. Cambridge, MA: Center for Information Systems Research, Sloan School of Management, MIT.

Malone, Thomas W. (1986) Computer Support for Organizations: Toward an Organizational Science. John M. Carroll (Ed.), Interfacing Thought: Cognitive Aspects of Human-Computer Interaction. Cambridge, MA: MIT Press.

Malone, Thomas W., Grant, Kenneth R., Lai, Kum-Yew, Rao, Ramana, and Rosenblitt, David. (1986) Semi-Structured Messages Are Surprisingly Useful for Computer-Supported Coordination. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Malone, Thomas W., Grant, Kenneth R., Turbak, Franklyn A., Brobst, Stephen A., and Cohen, Michael D. (1987) Intelligent Information Sharing Systems. *Communications of the ACM* **30**, 5, 390-402.

Malone, Thomas W., Yates, JoAnne, and Benjamin, Robert I. (1987) Electronic Markets and Electronic Hierarchies. *Communications of the ACM* **30**, 6, 484-497.

Mamrak, S.A., Kaelbling, M.J., Nicholas, C.K., and Share, M. (1987) A Software Architecture for Supporting the Exchange of Electronic Manuscripts. *Communications of the ACM* **30**, 5 (May, 1987), 408-414.

Marshall, Catherine C. (1987) Exploring Representation Problems Using Hypertext. Proceedings of Hypertext '87. Chapel Hill, NC: University of North Carolina.

Mayer, Richard E. (1987) Learning a Programming Language. John M. Carroll (Ed.), Interfacing Thought: Cognitive Aspects of Human-Computer Interaction. Cambridge, MA: MIT Press.

McAlear, James H., and Wehrung, John M. (1984) The Biochip: Now, 2000 A.D., and Beyond. *IBM* **15** (August, 1984), 51-56.

McKay, Charles W. (1987) A Proposed Framework for the Tools and Rules to Support the Life Cycle of the Space Station Program. Proceedings of the COMPASS Conference. New York: Association for Computing Machinery.

Meindl, James D. (1987) Chips for Advanced Computing. *Scientific American* **257**, 4 (October, 1987), 78-88.

Meyrowitz, N. (1986) Intermedia: The Architecture and Construction of an Object-Oriented Hypertext/Hypermedia System and Applications Framework. Proceedings OOPSLA 1986. New York: ACM.

Miller, James R. (1986) Human-Computer Interaction and Intelligent Tutoring Systems. Austin, TX: MCC Technical Report HI-086-17, Microelectronics and Computer Technology Consortium.

Minsky, Marvin. (1986) The Society of Mind. New York: Simon & Schuster.

Mohl, R.F. (1982) Cognitive Space in the Interactive Movie Map: An Investigation of Spatial Learning in Virtual Environments. Cambridge, MA: doctoral dissertation, Department of Architecture, MIT.

Mylopoulos J. (1986) On Knowledge Base Management Systems. Michael L. Brodie and John Mylopoulos, Eds, On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies. New York: Springer-Verlag.

Narayanaswamy, K., and Scacchi, W. (1987) Maintaining Configurations of Evolving Software Systems. *IEEE Transactions on Software Engineering* 13, 3, 324-334.

NASA. (1986a) Software Support Environment: Request For Proposal, 9-BF3-64-6-2P. Houston, TX: Johnson Space Center.

NASA. (1986b) Technical and Management Information System: Request For Proposal, 9-BF3-32-6-01P. Houston, TX: Johnson Space Center.

NASA. (1986c) Space Station Program Software Support Environment System, Environment and Characteristics Document. Houston, TX: Johnson Space Center.

NASA. (1986d) Space Station Program Definition and Requirements, Section 2: Program Management Requirements, Part 8: Documentation Requirements. Houston, TX: Johnson Space Center.

NASA. (1986e) Program Support Communications Requirements Document (PSCRD) Volume III, TMIS document Y8600029. Houston, TX: Johnson Space Center.

NASA. (1986f) Space Station Information System (SSIS) Definition Document, TMIS document Y8600050. Houston, TX: Johnson Space Center.

NASA. (1986g) NASA Strategic Automated Information Management (AIM) Plan, TMIS Document Y8600014. Houston, TX: Johnson Space Center.

NASA. (1987) Space Station Software Support Environment Functional Requirements Specification. Houston, TX: Johnson Space Center.

Nelson, Ted. (1987) Literary Machines. San Antonio, TX: Project Xanadu.

Nickerson, Raymond S. (1987) Productivity in the Space Station. Cambridge, MA: BBN Laboratories.

Norman, Donald A. (1987) Cognitive Engineering--Cognitive Science. John M. Carroll (Ed.), Interfacing Thought: Cognitive Aspects of Human-Computer Interaction. Cambridge, MA: MIT Press.

Office of Technology Assessment, U.S. Congress. (1981) Computer-Based National Information Systems: Technology and Public Policy Issues. Washington, DC: U.S. Government Printing Office.

Office of Technology Assessment, U.S. Congress. (1985a) Information Technology R & D: Critical Trends and Issues. Washington, DC: U. S. Government Printing Office.

Office of Technology Assessment, U.S. Congress. (1985b) International Cooperation and Competition in Civilian Space Activities. Washington, DC: U.S. Government Printing Office.

Parker, D., Carey, M., Golshani, F., Jarke, M., Sciore, E., and Walker, A. (1986) Logic Programming and Databases. Larry Kerschberg, Ed., Expert Database Systems: Proceedings From the First International Workshop. Menlo Park, CA: Benjamin/Cummings Publishing.

Parnas, David L. (1985) Software Aspects of Strategic Defense Systems. *American Scientist* **73**, 5 (September-October, 1985), 432-440.

Parnas, David L., and Clements, Paul C. (1986) A Rational Design Process: How and Why to Fake It. *IEEE Transactions in Software Engineering*, **SE-12**, 2 (February, 1986), 251-257.

Pea, Roy D. (1987) Human-Machine Symbiosis: Exploring Hypermedia as New Cognitive and Cultural Technologies. Proceedings of the International Symposium on Computers, Cognition, and Epistemology. Denmark: University of Aarhus.

Penedo, M.H., and Stuckle, E.D. (1985) PMDB--A Project Master Database for Software Engineering Environments. Proceedings of the Eighth International Conference on Software Engineering. New York: ACM.

Perrault, C. Raymond, and Grosz, Barbara J. (1986) Natural-Language Interfaces. J.F. Traub (Ed.), Annual Review of Computer Science, Volume 1. Palo Alto, CA: Annual Reviews.

Reichman, Rachel. (1986) Communication Paradigms for a Window System. Donald A. Norman and Stephen W. Draper (Eds.), User-Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Lawrence Erlbaum.

Rich, Charles, and Waters, Richard C., Eds. (1986) Readings in Artificial Intelligence and Software Engineering. Los Altos, CA: Morgan Kaufmann.

Rittel, Horst, and Webber, Melvin. (1973) Dilemmas in a General Theory of Planning. *Policy Sciences* **4**, 111-127.

Rouse, William B., and Morris, Nancy M. (1986) On Looking into the Black Box: Prospects and Limits in the Search for Mental Models. *Psychological Bulletin* **100**, 3 (March, 1986), 324-355.

Rudisill, M., Burns, M.J., and Warren, D.L. (1985) Expert and Nonexpert Search Task Performance with Current and Reformatted Space Shuttle Displays. Houston, TX: Human Factors Laboratory, Johnson Space Center.

Russell, Daniel M., Moran, Thomas P., and Jordan, Daniel S. (1987) The Instructional Design Environment. J. Psotka, L.D. Massey, and S.A. Mutter (Eds.), Intelligent Tutoring Systems: Lessons Learned. Hillsdale, NJ: Lawrence Erlbaum.

Sathi, Arvind, Morton, Thomas E., and Roth, Steven F. (1986) Callisto: An Intelligent Project Management System. *The AI Magazine* 6, 4 (Winter, 1986), 34-52.

Scacchi, Walt. (1987) The System Factory Approach to Software Engineering Education. R. Fairley and P. Freeman (Eds.), Educational Issues in Software Engineering. New York: Springer-Verlag.

Shneiderman, Ben, and Morariu, Janis. (1986) Design and Research on The Interactive Encyclopedia System (TIES). Proceedings of the 29th Conference of the Association for the Development of Computer-Based Instructional Systems. Washington, DC: ADCIS.

Shneiderman, Ben. (1987a) User Interface Design and Evaluation for an Electronic Encyclopedia. Proceedings of the Second International Conference on Human-Computer Interaction. New York: North-Holland.

Shneiderman, Ben. (1987b) Designing the User Interface: Strategies for Effective Human-Computer Interaction. Reading, MA: Addison-Wesley.

Shneiderman, Ben. (1983) Direct Manipulation: A Step Beyond Programming Languages. *IEEE Computer* 16, 8, 57-69.

Shasha, Dennis. (1985) NetBook: A Data Model to Support Knowledge Exploration. Proceedings of the Very Large Data Base Conference. New York: ACM.

Shasha, Dennis. (1986) When Does Non-Linear Text Help? Larry Kerschberg (Ed.), Expert Database Systems: Proceedings of the First International Workshop. New York: Benjamin Cummins.

Shaw, Mary. (1985) Beyond Programming-in-the-Large: The Next Challenges for Software Engineering. Annual Report of the Software Engineering Institute. Pittsburgh, PA: Software Engineering Institute, Carnegie-Mellon University.

Sheil, B.A. (1982) Coping with Complexity. Richard A. Kasschau, Roy Lachman, and Kenneth R. Laughery (Eds.), Information Technology and Psychology: Prospects for the Future. New York: Praeger.

Smith, Randall B. (1987) Experiences with the Alternate Reality Kit: An Example of the Tension Between Literalism and Magic. Proceedings of CHI+GI 1987. New York: ACM.

Stefik, Mark. (1986) The Next Knowledge Medium. *The AI Magazine* 7, 1 (Spring, 1986), 34-46.

Stefik, Mark, Foster, Gregg, Bobrow, Daniel G., Kahn, Kenneth, Lanning, Stan, and Suchman, Lucy. (1987) Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings. *Communications of the ACM* 30, 1, 32-47.

Straussman, Paul. (1985) Information Payoff: The Transformation of Work in the Electronic Age. New York: Free Press.

Suchman, Lucy A. (1985) Plans and Situated Actions: the Problem of Human-Machine Communication. Palo Alto, CA: Technical Report ISL-6, Xerox Palo Alto Research Center.

Szolovits, Peter. (1985) Types of Knowledge as Bases for Reasoning in Medical AI Programs. I. De Lotto and M. Stefanis (Eds.), Artificial Intelligence in Medicine. Amsterdam: Elsevier.

Szolovits, Peter. (1986) Knowledge-Based Systems: A Survey. Michael L. Brodie and John Mylopoulos (Eds.), On Knowledge Base Management Systems: Integrating Artificial Intelligence and Database Technologies. New York: Springer-Verlag.

Taylor, James C. (1987) Job Design and the Quality of Working Life. In Robert E. Kraut (Ed.), Technology and the Transformation of White-Collar Work. Hillsdale, NJ: Lawrence Erlbaum.

Trigg, Randall H. and Weiser, Mark. (1986) Textnet: A Network-Based Approach to Text Handling. *ACM Transactions on Office Information Systems* **4**, 1 (January, 1986), 1-23.

Trigg, Randall H., Suchman, Lucy A., and Halasz, Frank G. (1986) Supporting Collaboration in NoteCards. Proceedings of the 1986 Conference on Computer Supported Cooperative Work. Austin, TX: Microelectronics and Computer Technology Corporation.

Tsunoda, Yoshito. (1985) Present Status And Future Trends In Erasable Optical Memories. Proceedings of the Laser and Electro Optics Society. Washington, DC: Laser and Electro Optics Society.

Tullis, Thomas S. (1987) Screen Design. M. Helander (Ed.), Handbook of Human-Computer Interaction. Amsterdam: North-Holland.

Tullis, T.S. (1987) A System for Evaluating Screen Formats: Research and Application. H.R. Hartson and D. Hix (Eds.), Advances in Human-Computer Interaction, Volume 2. Norwood, NJ: Ablex.

Turkle, Sherry. (1984) The Second Self: Computers and the Human Spirit. New York: Simon & Schuster.

Wenger, Etienne. (1987) Artificial Intelligence and Tutoring Systems: Computational Approaches to the Communication of Knowledge. Los Altos, CA: Morgan Kaufmann.

Weyer, Stephen A. and Borning, Alan H. (1985) A Prototype Electronic Encyclopedia. *ACM Transactions on Office Systems* **3**, 1, 63-88.

Wilbur, Ken, Ed. (1982) The Holographic Paradigm. Boston, MA: New Science Library.

Wilson, Kathleen S. (1987) The Palenque Optical Disc Prototype. New York: Technical Report No. 44, Bank Street College of Education.

Winograd, Terry, and Flores, Fernando. (1986) Understanding Computers and Cognition: A New Foundation for Design. Norwood, NJ: Ablex.

Woods, David D. (1986) Cognitive Technologies: the Design of Joint Human-Machine Cognitive Systems. *The AI Magazine* **6**, 4 (Winter, 1986), 86-92.

Woods, William A. (1975) What's in a Link: Foundations for Semantic Networks. Brachman, Ronald J. and Levesque, Hector J., Eds. Readings in Knowledge Representation. Los Altos, CA: Morgan Kaufmann.

Yankelovitch, N., Meyrowitz, N., and van Dam, A. (1985) Reading and Writing the Electronic Book. *Computer* **18**, 10 (October, 1985), 15-30.

Yudkin, Howard L., and Nidiffer, Kenneth E. The Software Productivity Consortium. Reston, VA: Software Productivity Consortium.

Appendix: Functional Requirements for TMIS and SSE

Technical and Management Information System (TMIS) Overview

TMIS is an SSP-wide system which will link all management levels of NASA, all international partners, SSP contractors, other NASA organizations and NASA customers through a system offering a comprehensive, yet standardized set of information management technologies and procedures [NASA, 1986d]. TMIS goals and objectives include an automated environment for the evolution of the SSP which integrates the principal participants, their needs, and requirements; minimizes waste; maximizes resources; and interfaces with external NASA systems to create a common research, development and operations environment throughout the life cycle of the SSP.

To better understand how TMIS will support the development of the SSP, following are twenty-eight processes which have been identified as critical[NASA/JSC, 1986b]:

SSP PROCESSES

1. Design
2. Technical Analysis
3. Design Review
4. Requirements Analysis
5. Configuration Management
6. Documentation
7. Interface Control
8. Planning
9. Maintenance
10. Policy Development
11. Technical Contract Mgt.
12. Sched/Project Mgt.
13. Budgeting
14. Integration
15. Cost/Financial Analysis
16. International Relations
17. Training
18. Administration
19. Admin. Contract Mgt.
20. Performance Mgt.
21. Operations
22. Prototyping
23. Program Review
24. Acquisition
25. External Affairs
26. Test & Verification
27. Inventory Mgt.
28. Customer Relations

To implement these twenty-eight SSP processes, TMIS will contain sixteen fundamental capabilities in the form of automated tools. These include:

AUTOMATED TMIS CAPABILITIES

1. A Common User Interface
2. Computer Aided Engineering
3. Remote System Access
4. Image Processing
5. Word Processing
6. Electronic Mail
7. Spreadsheets
8. Scheduling/Calendaring
9. Computer Aided Design
10. Data Base Management Sys.
11. File Transfer
12. Cost/Price Modeling
13. Graphics
14. Teleconferencing
15. Project Management
16. Computer Aided Instruct.

These tools or capabilities are intended to reflect the general TMIS requirements of:

- Ease of Use
- Maintainability
- High Performance Level
- Expandability
- Reliability
- Security
- Integration

Both SSP and TMIS are planned to evolve over phased implementation. Phase 1 of TMIS has recently been completed; it supported the preliminary design phase of the SSP. Phase 2 of the TMIS project is currently active and spans an 8 year interval (to mid-1995, which corresponds to a timeframe which just precedes planned actual operation of the Space Station). TMIS Phase 2 consists of six "Initial Operating Configurations" or IOC: IOC' (the initial phase) through IOC-5 extend from the present through 1995. The "Evolutionary Phase" extends from the conclusion of Phase 2 into the future and encompasses space station operation and sustainability. In the late summer of 1987, the TMIS Request-For- Proposal [NASA, 1986b] was awarded to Boeing Computer Services as the prime contractor, thus initiating the start of TMIS Phase 2.

To better understand the scalar component of the TMIS system, below are profiles of the TMIS user base and the information volumes expected to occur from IOC' through IOC-5. It should be noted that these estimates are probably lower than will actually occur. The twenty-eight SSP processes are expected minimally to yield the following information volumes in each incremental IOC. It is plausible to assume these estimates reflect core information of SSP development and that a significantly greater volume of secondary supporting information will also be generated.

TMIS INFORMATION VOLUME ESTIMATES

Time	IOC	Megabytes per IOC	Running Total
04/87	IOC'	141,319.7 megs	141,319.7
12/87	IOC-1	207,285.8 megs	348,605.5
12/88	IOC-2	313,638.9 megs	662,244.4
05/89	IOC-3	500,509.2 megs	1,162,753.6
09/90	IOC-4	645,274.4 megs	1,808,028.0
09/92	IOC-5	740,550.1 megs	2,548,578.1
12/94	Evol	n/a	n/a

NOTE: Not included in these estimates is the information generated in Phase 0 and Phase 1, which encompassed a period from approximately 1984 to IOC'.

The size of the TMIS user population has been estimated in two year increments over the TMIS Phase 2 time period. The number of actual TMIS users will be significantly greater than specified here, as the following estimates are based upon a Full Time Equivalent (FTE) projection. (Civil Service projections encompass NASA SSP manpower at levels A, B, C, and D. These levels represent SSP management structure where "A" is program direction or NASA HQ, "B" is program management or requirements and conceptual design, "C" is project management or the work packages assigned to field centers, "D" is component development and design, and "E" is contractor elements. Recently these designations were expanded with a numerical format where level "A" is equal to 1, "B" equal to 2, and "C" equal to 3.)

ESTIMATED TMIS USER POPULATION

(in Full Time Equivalents)

	1987	88	90	92	94
Civil Service	150	212	416	426	398
Prime Contractors	17	136	258	192	64
Non Prime Contracts.	22	110	151	130	50
TOTAL FTEs	189	458	825	748	512
Estimated Number of TMIS Users (Non-FTE)	2007	4616	11525	9866	5482

NOTE: Time intervals provided do not map directly to IOC phases.

Direct labor projections for development and implementation of TMIS over the Phase 2 period have been budgeted at between 3,981,312 and 4,104,445 hours. Additionally, NASA has retained an option to increase performance of the TMIS prime contractor during the course of the development project and has budgeted 820,889 hours in the event this option is exercised. This translates into 1990 to 2052 person years (2000 hours = 1 person year) of effort for contracted work outside the NASA environment.

A basic component of the TMIS strategy is the concept of workstations dedicated to the TMIS network. Workstations oriented to three types of functions have been identified as a basic necessity, although ideally any segment of TMIS could be accessed from any type of TMIS workstation. Preliminary workstation configuration for the major NASA entities and Space Station levels A and B (administration and design levels) is estimated for IOC as follows:

TMIS WORKSTATION ESTIMATE

Management Workstation	Documentation Workstation	Engineering Workstation
180	160	310

TMIS Phase 1 was designed to provide a basic environment in which early conceptualization of the SSP could evolve. Phase 1 functionality was not intended to be carried over to the current Phase 2 unless there were clear, long term advantages to doing so. Phase 1 provided solutions on a non-integrated basis for six basic processes (applications):

TMIS PHASE 1 PROCESSES

Applications	Solution
Electronic Mail	IBM Profs System
Document Generation/Storage	IBM Displaywrite 3
Scheduling/Project Mgt.	Artemis System
Data Management	RIM 7 and ADABASE
CAD/CAE	DEC based IDEAS**2
Budgeting	NOMAD DBMS

As of the end of 1986, current levels of information volumes generated during the Phase 1 period within each of the six processes defined above are estimated as follows:

CURRENT TMIS INFORMATION VOLUME

Electronic Mail	Unavailable
Document Generation	3 gigabytes
Scheduling	Unavailable
Data Management	250 megabytes (plus 2 gigabytes of secondary information)
CAD/CAE	400 megabytes
Budgeting	Unavailable

TMIS Phase 1 baseline software for individual workstations (personal computers) included:

Database Mgt.	R:base 5000 (or later)
Word Processing	Displaywrite 3
Asynchronous Comm.	Crosstalk, Smartcom, SIMPC
Bisynchronous Comm.	TBD
Operating System	MS-DOS 2.0 (or later)
Spreadsheet	Lotus 1-2-3

The TMIS Request For Proposal outlines the basic requirements for documentation workstation capabilities to be implemented over the Phase 2 period. Summarized, these requirements call for creating, editing, storing, retrieving, searching, distributing, and printing multi-media documents. (A "multi-media" or "mixed object" format contains documents composed of text, numbers, graphics, and images in which the various information media are not treated as separate files, but are all part of a common document file architecture.)

Software Support Environment (SSE) Overview

The SSE System consists of hardware, computer networks, software, procedures, standards, hardware specifications, documentation, policies, and training materials forming an integrated whole. In the context of the SSP, SSE is the set of geographically distributed elements that, when operated as a system, support the life cycle management of all SSP operational software [NASA, 1986a].

The justification for developing the SSE System for the SSP stems from three important issues:

1. Space mission software costs as a percentage of overall costs are continually increasing, and the majority of the costs are in the sustaining engineering of that software.
2. The SSE operating in conjunction with the SSIS insures a migration path and compatibility for software developed during Phase 2 design and development to an operational status for the Space Station.
3. By creating a common software development facility and standardized policy and procedures to minimize integration barriers, the SSE supports the design-to-cost strategy of SSP and coordinates software development among the participating field centers.

This System is composed of five major task areas {Work Breakdown Structures (WBS) 1-5} envisioned to support the development, integration and test of all Space Station software across NASA centers, international participants, and SSP contractors.

SSE will evolve over time, with requirements analysis and preliminary design scheduled to begin in mid-1987 and completion scheduled for mid-1993.

A sustaining engineering phase will then be initiated to support space station operation. A high level objective of SSE is the concept of "transparency to change" from the perspective of its users for the complete life cycle of the SSP, a period of at least 25 years. This is to be accomplished by loosely coupling the hardware, software and software engineering methodologies as much as possible to Commercial-Off-The-Shelf technology (COTS).

The three major elements of the SSE are the SSE Development Facility (SSEDF), the SSP Software Production Facilities (located at each major NASA field center), and the Integration Facility [NASA, 1986c]. The SSEDF is used by the Space Station Program for the life cycle management of the SSE. This element is composed of automated data processing electronic (ADPE) hardware and system software, interfaces to computer networks, unique hardware and a proper subset of the SSE.

A SSP Software Production Facility is any SSE System Element that will be used by a NASA center, international participant, SSP contractor, or SSP user for the life cycle management of SSP flight or ground operational software. A SSP Software Production Facility is composed of ADPE hardware and system software, interfaces to computer networks, unique hardware, and a proper subset of the SSE.

The Integration Facility is some SSP Software Production Facility that will be used by the SSP for integrating, acceptance testing, and delivery of all SSP flight operational software prior to transmission to the on-board Data Management System (DMS). At appropriate schedule points, all flight software will be delivered to the Integration Facility to be interlinked.

The development of the SSE has been budgeted to required approximately 1,650,240 direct labor hours of effort. This translates to 825 labor years (2000 hrs. = 1 year). The Ada programming language has been selected as the baseline language to be used for the development of all operational software for the Space Station Program and the SSE. (Ada is a registered trademark of the U.S. Government, Ada joint programming office.)

The following topics represent both the philosophy and the conceptual requirements upon which the design of the SSE System is based [NASA, 1986a]:

- Commonality
- Expandability
- Subsetability/Modularity
- Replicability
- Human Factors
- Consistency
- Performance
- Standardization

Functionally, the SSE provides four areas of support to users:

- a. Software Configuration Management
- b. Software-related scheduling and project management
- c. Document processing embedded in the software tools
- d. Data Base Management tools supporting the software tools

SSE is directly interfaced with the SSIS (for migration of operational flight software) and with the PSCN, which will function as the communications network for SSE and TMIS.

SSE Documentation Requirements

Documentation is a process that is required by all phases of the software life cycle. As such, the SSE Document Processing Element will provide tools that facilitate and encourage the generation of complete documentation at each step in the software life cycle process. In addition, all documentation produced by such tools will be in a form that is compatible with electronic storage in SSE databases and usable as input by other tools and capabilities in the SSE. The SSE Document Processing Element will provide for the full integration of text and graphics. This means that computer-generated and computer-processed images can be freely intermixed with text during any phase of document processing.

Documentation for the SSE and the elements of the SSE project is substantial. To illustrate, a small sampling of the types of documentation required for the SSE within the early WBS periods is delineated below [NASA, 1987].

SSE DOCUMENTATION REQUIREMENTS

Management Plan
Engineering & Integration Plan
Software Development Plan
Sustaining Engineering Plan

Interface Control Plan
Interface Requirements Document
Interface Control Document
Element Performance Agreement

Preliminary Design Document
Detailed Design Document

Operations Manual

Standards Document
SR&QA Plan
Configuration Management
Training Plan

Concept Document
Functional Requirement Specs
Detailed Requirements Specs

Contract End Item List
Version Description Document

Development Test Plan

User Guide
Methods Manual
Programmers Manual
Development Test Specs
Development Test Procedures
Development Test Reports

System Element Integration Test Spec
System Element Integration Test Procedure
System Element Integration Test Report for SSEDF

Contract End Item List
Version Description Document
Prototype Demonstration Report
Prototype Plan
Prototype Description Document

IV&V Test Plan
IV&V Test Specification
IV&V Test Procedure
IV&V Test Report
Cost Report
System Element Status Report
Status Report
Lessons Learned Report
Review Minutes Report

Element Metrics Report
SR&QA Report
Training Report
Acceptance Test Plan
Acceptance Test Report
Element Acceptance Test Report

This list depicts one overarching level of documentation requirements for the SSE. Correspondingly more detailed documentation is required for each SSE System Element and SSE interfaces to other Space Station systems. The result is massive volumes of multi-media information which must be electronically created, revised, integrated, stored, searched, retrieved, and disseminated throughout a large user base in varying geographic locations over an extended period of time.